Smart Innovation, Systems and Technologies 153

Debahuti Mishra
Rajkumar Buyya
Prasant Mohapatra
Srikanta Patnaik *Editors*

Intelligent and Cloud Computing

Proceedings of ICICC 2019, Volume 2

KES International

Springer

Smart Innovation, Systems and Technologies

Volume 153

Series Editors

Robert J. Howlett, Bournemouth University and KES International, Shoreham-by-sea, UK

Lakhmi C. Jain, Faculty of Engineering and Information Technology, Centre for Artificial Intelligence, University of Technology Sydney, Sydney, NSW, Australia

The Smart Innovation, Systems and Technologies book series encompasses the topics of knowledge, intelligence, innovation and sustainability. The aim of the series is to make available a platform for the publication of books on all aspects of single and multi-disciplinary research on these themes in order to make the latest results available in a readily-accessible form. Volumes on interdisciplinary research combining two or more of these areas is particularly sought.

The series covers systems and paradigms that employ knowledge and intelligence in a broad sense. Its scope is systems having embedded knowledge and intelligence, which may be applied to the solution of world problems in industry, the environment and the community. It also focusses on the knowledge-transfer methodologies and innovation strategies employed to make this happen effectively. The combination of intelligent systems tools and a broad range of applications introduces a need for a synergy of disciplines from science, technology, business and the humanities. The series will include conference proceedings, edited collections, monographs, handbooks, reference books, and other relevant types of book in areas of science and technology where smart systems and technologies can offer innovative solutions.

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

**** Indexing: The books of this series are submitted to ISI Proceedings, EI-Compendex, SCOPUS, Google Scholar and Springerlink ****

More information about this series at http://www.springer.com/series/8767

Debahuti Mishra · Rajkumar Buyya ·
Prasant Mohapatra · Srikanta Patnaik
Editors

Intelligent and Cloud Computing

Proceedings of ICICC 2019, Volume 2

Springer

Editors
Debahuti Mishra
Department of Computer Science and Engineering
Siksha 'O' Anusandhan Deemed to be University
Bhubaneswar, Odisha, India

Prasant Mohapatra
Department of Computer Science
University of California
Davis, CA, USA

Rajkumar Buyya
Cloud Computing and Distributed Systems Laboratory
University of Melbourne
Melbourne, VIC, Australia

Srikanta Patnaik
Department of Computer Science and Engineering
Siksha 'O' Anusandhan Deemed to be University
Bhubaneswar, Odisha, India

ISSN 2190-3018 ISSN 2190-3026 (electronic)
Smart Innovation, Systems and Technologies
ISBN 978-981-15-6201-3 ISBN 978-981-15-6202-0 (eBook)
https://doi.org/10.1007/978-981-15-6202-0

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

Preface

The 1st International Conference on Intelligent and Cloud Computing (ICICC 2019) was organized by the Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan (Deemed to be University) during 16–17 December 2019. ICICC 2019 has provided a suitable platform for academicians, practitioners, researchers and experts to showcase their work and findings as well as exchange ideas and future directions while sharing their experiences with each other. ICICC 2019 has focused on recent research trends and advancements in cloud computing and its importance in diversified application fields. Intelligent and cloud computing integrated with other relevant techniques not only adds values to crucial decisions but also provides new direction to advancement in key technical areas such as adoption of Internet of things, distributed computing, cloud computing, blockchain and big data. It provides significant insights that support managers in various sectors such as human resource developments, resource management, biological data processing, marketing strategies, supply chain management, logistics, business operations and financial markets, thus making it a potential research area. The conference has received a good response with a large number of submissions. The total number of relevant papers accepted for publication has been broadly divided into five major tracks including: (i) intelligent and cloud computing, (ii) software engineering, (iii) wireless sensor network, (iv) IoT and its application and (v) AI and machine learning.

The first three tracks, namely intelligent and cloud computing, service-oriented software engineering and wireless sensor networks (WSN), have been included in Volume I of the book.

The fourth part, Internet of Things (IoT), is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IoT has evolved with a greater generation of data. Internet of things cloud service creates excessive communication between inexpensive sensors in the IoT which means even greater connectivity; billions of connected devices and machines will soon join human users.

In the last part, Artificial Intelligence (AI) is defined as the science of the engineering that makes the machine intelligent. The term AI has been modernized to include machine intelligent, natural sciences, natural computation, swarm intelligence, neural networks and fuzzy logic, and it moves from building systems that are intelligent to intelligent systems. Several factors have affected and are affecting the revolution of AI, such as the data-driven products, new platforms and the maturity of machine learning supported by cloud computing technology.

ICICC 2019 has not only encouraged the submission of unpublished original articles in the field of intelligent and cloud computing issues but also considered the several cutting-edge applications across organizations and firms while scrutinizing the relevant papers.

Bhubaneswar, India Debahuti Mishra
Melbourne, Australia Rajkumar Buyya
Davis, USA Prasant Mohapatra
Bhubaneswar, India Srikanta Patnaik

Contents

About the Editors

Prof. (Dr.) Debahuti Mishra received her B.E. degree in Computer Science and Engineering from Utkal University, Bhubaneswar, India, in 1994; her M.Tech. degree in Computer Science and Engineering from KIIT Deemed to be University, Bhubaneswar, India, in 2006; and her Ph.D. degree in Computer Science and Engineering from Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India, in 2011. She is currently working as a Professor and Head of the Department of Computer Science and Engineering, at the same university.

Dr. Rajkumar Buyya is a Redmond Barry Distinguished Professor and Director of the Cloud Computing and Distributed Systems (CLOUDS) Laboratory at the University of Melbourne, Australia. He is also the founding CEO of Manjrasoft, a spin-off company that commercializes university innovations in cloud computing. He served as a Future Fellow of the Australian Research Council from 2012 to 2016. He has authored over 625 publications and seven textbooks, including "Mastering Cloud Computing", published by McGraw Hill, China Machine Press, and Morgan Kaufmann for Indian, Chinese and international markets, respectively.

Dr. Prasant Mohapatra is the Vice Chancellor for Research at the University of California, Davis. He is also a Professor at the Department of Computer Science and served as the Dean and Vice-Provost of Graduate Studies at the University of California, Davis, from 2016 to 18. He was also an Associate Chancellor in 2014–16, and the Interim Vice-Provost and CIO of UC Davis in 2013–14. Further, he was the Department Chair of Computer Science from 2007 to 13 and held the Tim Bucher Family Endowed Chair Professorship during that period. He has also been a member of the faculty at Iowa State University and Michigan State University.

Dr. Srikanta Patnaik is a Professor at the Department of Computer Science and Engineering, Faculty of Engineering and Technology, SOA University, Bhubaneswar, India. Dr. Patnaik has published 100 research papers in international journals and conference proceedings. Dr. Patnaik is the Editor-in-Chief of the International Journal of Information and Communication Technology and

International Journal of Computational Vision and Robotics, published by Inderscience Publishing House, England, and also Editor-in-Chief of a book series on “Modeling and Optimization in Science and Technology”, published by Springer, Germany.

Internet of Things (IOT)

An Energy-Efficient Data Routing in Weight-Balanced Tree-Based Fog Network

Sayantan Saha and Arnab Mitra

Abstract Investigation for the scope for implementation of green computing in fog network is an important focus among researchers. For this reason, a weight-balanced tree-based architecture in fog network is presented in this research. It is observed that the proposed design is having an advantage for the possession of a fixed (or, almost fixed) length routing table, which further ensures the simplicity of design and fast execution. A detailed analysis discovers the inherent energy efficiency of proposed architecture towards data routing, which further explores the potential of proposed architecture towards the implementation of green computing in fog network.

Keywords Fog networks · Tree-based architecture · Weight-balanced tree · Data routing · Energy efficiency · Green computing

1 Introduction

Advancements of information technology in view of hardware, software and firmware have helped the earlier human society to be evolved into a high-tech digital human society. Today, the human society is greatly blessed with cloud computing-based applications. Present days, the fog computing is a buzzing and new technology, which may be considered an advancement of existing cloud technology. Thus, enhancements and advancements in fog computing have gained attentions among researchers [1].

Several research areas may be explored towards the advancements of fog computing. In our studies, we found data routing in fog computing may be further enhanced to facilitate low-power-driven implementation of fog networks. Unfortunately, in our studies, we found that the said research gap was not addressed. For

S. Saha · A. Mitra (✉)
Department of Computer Science & Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: arnabmitra@soa.ac.in

S. Saha
e-mail: sayantansaha@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_1

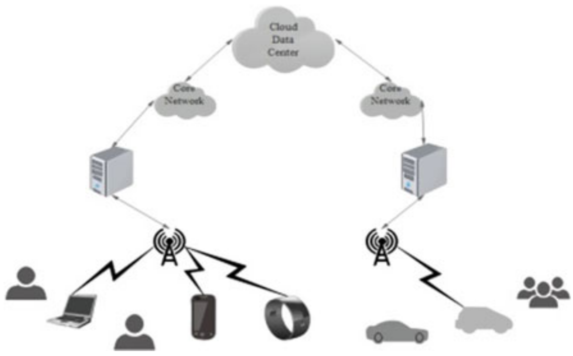


Fig. 1 Typical fog computing network

this reason, an enhanced fog computing architecture is presented in this research, which in turns facilitates energy-efficient data routing. Brief discussions on several key terms with a connection to our presented research are presented in the following subsections.

1.1 Fog Computing

With advancement of Internet, the distributed computing of past has transformed into the cloud computing, which has further evolved into fog computing (also known as edge computing), where the network core is typically connected to the network edges to facilitated computing at edges as well as server. A typical fog computing network architecture is presented in Fig. 1.

1.2 Tree-Based Architecture

Tree is considered as well-established nonlinear data structure to facilitate the modelling of a nonlinear architectural design. Among several others, presented research focuses on two important tree structures, i.e. height-balanced tree and weight-balanced tree, which are briefly introduced in the following subsections.

1.2.1 Height-Balanced Tree

In each interior node, the height of the right sub-tree and left sub-tree differs by at most one [2]. A diagrammatic representation towards the explanation for the height-balanced tree is presented in Fig. 2.

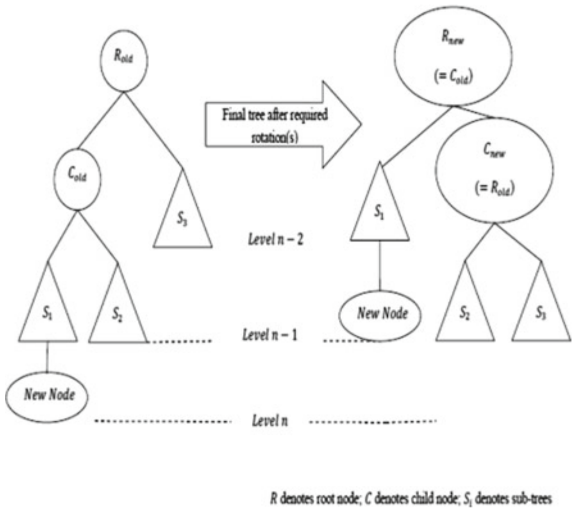


Fig. 2 A typical diagram for a height-balanced tree

1.2.2 Weight-Balanced Tree

The weight of right and left sub-tree in each node differs by at most one [2]. A diagram to explain the weight-balanced tree is presented in Fig. 3.

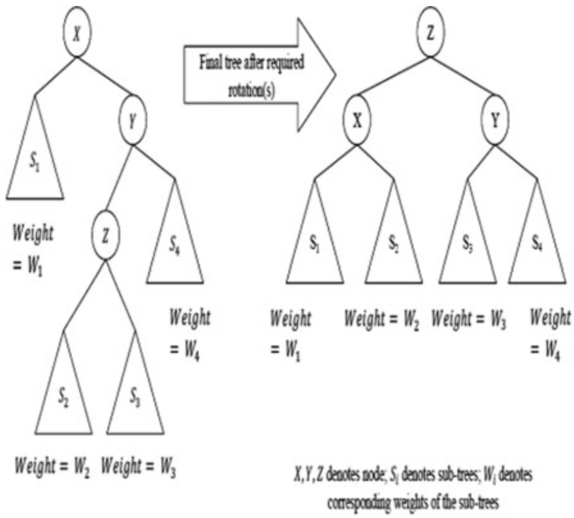


Fig. 3 Typical diagram for a weight-balanced tree

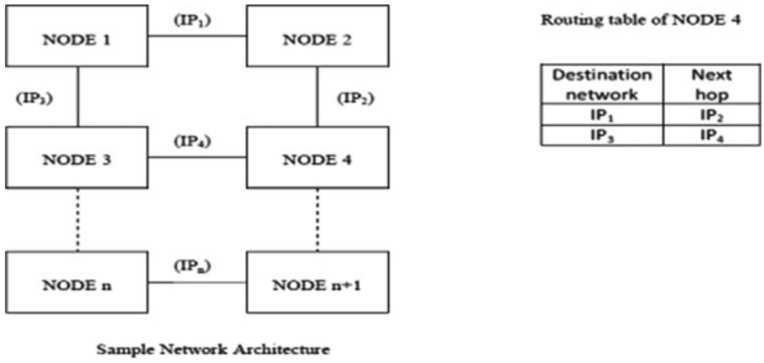


Fig. 4 Sample of a routing table

1.3 Data Routing

Data routing in a network is referred to as the movement/transmission of data packets over an Internet Protocol (IP), which is maintained through a routing table (often viewed in table format) and used by any IP-enabled devices (including routers and switches) [3, 4]. The importance of a routing table in view of a security attack was presented in [4]. A sample structure of a data routing table is presented in Fig. 4.

1.4 Green Computing

Green computing is viewed as the environmentally accountable process which helps towards the uses of any energy-consuming systems (e.g. computers) in an eco-friendly way to ensure reduced environmental impact by facilitating low energy consumption [5].

Due to limitation of pages for this research paper, key technologies related to this research (i.e. fog computing, tree-based architecture, data routing and green computing) are just introduced in a very brief way. Reader(s) of this paper is (are) requested to look into the following references to have detailed insights about fog computing [6, 7], height- and weight-balanced tree [2], routing [3, 4] and green computing [5].

The key features of this research paper are as follows.

1. An enhanced tree-based fog computing architecture is presented in this research which involves weight-balanced tree;
2. proposed weight-balanced tree-based fog computing architecture facilitates easy maintenance of routing table of fixed (or, almost fixed) size;
3. a theoretical investigation is presented to enhance energy-efficient data routing in fog computing;

4. low energy requirement is predicted towards the data transmission in proposed architecture of fog network.

Rest of the paper is organized as follows. A brief discussion on important past research works is presented in Sect. 2; our proposed architecture towards fog network is presented in Sect. 3; detailed discussions related to our proposed architecture are presented in Sect. 4; concluding remarks along with a possible future research direction are briefly presented in Sect. 5.

2 Related Works

Significant efforts may be found in the literatures towards the reliability and quality enhancements of existing cutting-edge computing technologies. Such efforts include researches on cloud computing along with several other areas. Among several others, a possible future research direction for cloud computing technology was investigated by Varghese et al. and presented in [1]. The advances in cloud computing and Internet of Things (IoT) have further emerged into fog computing. A brief introductory discussion on fog computing and its role was presented in [3, 6, 7]. Different data transmission policies and associated security threats in fog network were discussed in [3, 4]. Further, the importance of routing table in any fog network in view of data transmission and security enhancement was presented in [4].

On the other hand, environment sustainability is one crucial criterion for today's researchers and consultants, which results in low power consumption. For this reason, one important focus is found towards implementation of green computing and green clouds [5]. Towards the implementation of green computing, several energy-aware scheduling algorithms in distributed computing using cellular automata (CA) were described in detail in [8–10]. In different researches over time, the role of energy-efficient design and consideration of energy stability towards the modelling of green clouds were presented in [11, 12]. Further, a detailed investigation towards the efficiency in view of the implementation of green computing towards the optimization problem in fog networks was presented in [13]. It was concluded in [13] that "Randomized Algorithm AO-MAP and Approximation Scheme APO-MAP" [13, 14] were fit for possible consideration towards multi-application provisioning (MAP) and single application provisioning (SAP) in fog networks while ensuring green computing nature.

In another research, height-balanced *k*-ary tree-based architecture in fog computing was introduced in [15] to facilitate energy efficiency. In our studies, we found that the effective and improved architecture of fog network may result towards the improved/enhanced energy-efficient fog computing model. Though, several approaches may be found in the literatures towards the improvements of fog network to be evolved as a green model, we have not found architecture in fog network that targets simple and cost-effective routing table towards data transmission

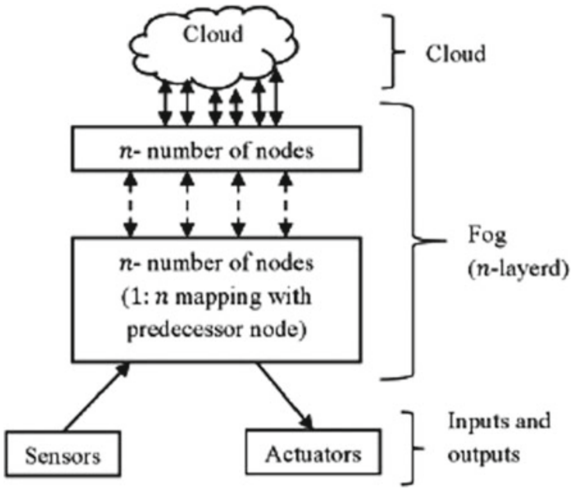


Fig. 5 Design of a height-balanced tree-based fog network as inspired from [15]

in fog networks. The typical architecture of a fog network as presented in [15] is presented once again in Fig. 5.

Unfortunately, we found the effective maintenance of routing table towards the data transmission was not described in [15]. For this reason, in presented research, a different tree-based (i.e. weight-balanced tree-based) architecture in fog computing is proposed to facilitate easy and effective of routing table towards data transmission. Proposed architecture is presented in the following Sect. 3.

3 Proposed Architecture Towards Fog Networks

As introduced and discussed in Sect. 2, we found the existing literature on energy-efficient fog network uses height-balanced-based *k*-ary tree [15]. We found that the proposed fog network of [15] is advantageous with reference to auto-maintenance of the network (tree) depth, as the architecture is based on the height-balanced tree. Though, a detailed investigation towards the green computing aspect of projected height-balanced tree-based architecture of [15] was presented, we found the data transmission and related routing table maintenance might be overlooked. We found, the ever-changing number of nodes at each level of the fog network may cause for a frequent nonuniform update(s) for the entries in corresponding routing table. Further, it may happen that the number of nodes at each level of the proposed height-balanced tree may quickly vary with a change (with insertion or removal of new node(s)/sub-network(s)), as the primary target of any height-balanced tree is to keep the depth of the existing network at fixed or almost fixed (refer height-balanced tree in [2]). Thus, a fixed-sized routing table may not exist.

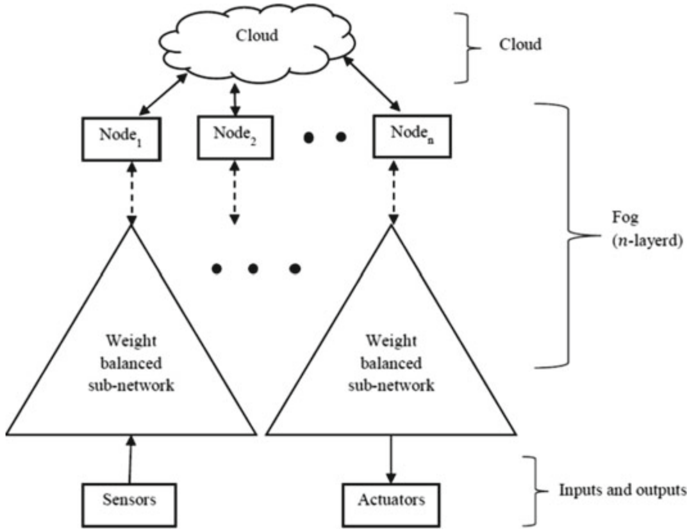


Fig. 6 Proposed design of a weight-balanced tree-based fog network

For this reason, to deal with the problem, we propose for weight-balanced tree-based architecture in fog network. Now, it will be easy to maintain the routing table as for each possible insertion or removal of new node(s)/sub-network(s) in existing network will result for no change/almost no change for the number of entries in routing table (refer weight-balanced tree in [2]). Hence, the creation and maintenance of a fix-sized routing table may be found in proposed weight-balanced tree-based fog network. A schematic diagram of proposed weight-balanced tree-based fog network is presented in Fig. 6.

The architecture as presented in Fig. 6 involves the concept of weight-balanced k-ary tree-based architecture in fog network instead of height-balanced tree as shown in [15]. For this reason, a change in network results towards modified architecture which once again contains weight-balanced sub-networks (trees). Thus, the routing table which is responsible for data transmission in network contains fixed (or almost fixed) population.

4 Discussions

It is discussed in Sect. 3 that the proposed weight-balanced tree-based architecture in fog network results towards the fixed- (or almost fixed)sized routing table. It was already analysed and presented in [16] that the fixed (or almost fixed) routing table helps towards the design simplicity and reduced time for processing. For this reason, proposed architecture enables simple design and fast processing. Hence, proposed weight-balanced tree-based design is advantageous in view of data transmission in network.

A theoretical approach towards the investigation for energy consumption and inherent energy efficiency (if any) for our proposed architecture are presented next.

Let us consider, the energy consumption per unit time for the data transmission in fog network using routing table is $E_{\mathrm{trans}_{\mathrm{Fog}}}$, the energy consumption per unit time for a processing node in fog network at active state is $E_{\mathrm{actv}_{\mathrm{processing-node}}}$, and the energy consumption per unit time for the said processing node of fog network at inactive (idle) state is $E_{\mathrm{idle}_{\mathrm{processing-node}}}$. Say there exists total n number of processing nodes in fog network, among which m (and $m < n$) nodes are involved in the data transmission session over a span of time t.

It was earlier analysed and reported that energy required for a processing node at active state is always more as compared to the energy requirement at idle state [8–10]. Hence, we have $E_{\mathrm{actv}_{\mathrm{processing-node}}} > E_{\mathrm{idle}_{\mathrm{processing-node}}}$.

Now, the total energy consumption $\left(E_{\mathrm{total}_{\mathrm{Fog}}}\right)$ for a data transmission over the said span of time t may be achieved using the following Eq. (1).

$$E_{\mathrm{total}_{\mathrm{Fog}}} = \left(m\left(E_{\mathrm{actv}_{\mathrm{processing-node}}}\right) + (n-m)\left(E_{\mathrm{idle}_{\mathrm{processing-node}}}\right)\right)t \tag{1}$$

It is already presented that the execution time is always shorter for data transmission using a fixed length routing table over data transmission using a variable length routing table [16]. Hence, the said span of time t for data transmission in Eq. (1) involving fixed length routing table in proposed weight-balanced tree-based fog network always should have less value in comparison to time required for a variable length routing table. Hence, in view of the routing based on routing table, the total energy consumption $E_{\mathrm{total}_{\mathrm{Fog}}}$ for weight-balanced tree-based fog network is low compared to height-balanced tree-based fog network. Thus, the proposed architecture facilitates energy-efficient routing in fog network.

5 Conclusions and Future Work

An enhanced tree-based approach involving weight-balanced k-ary tree-based architecture towards data transmission in fog network was presented in this research, and its advantages were explored further. It was explored that proposed approach is benefited with fixed (or almost fixed) length routing table which further ensured a simple routing table and its fast processing [16]. Further it was analysed that faster execution of fixed (or almost fixed) length routing table facilitates low energy consumption in view of data transmission. Thus, the proposed approach exhibited its potential towards the modelling of energy-efficient routing in fog network.

In future, we further plan to extend this research to have a detailed investigation about the inherent advantages (if any) for weight-balanced tree-based modelling towards energy-efficient routing in fog network.

Acknowledgements The authors acknowledge sincerely the anonymous reviewers for their useful suggestions.

References

1. Varghese, B., Buyya, R.: Next generation cloud computing: New trends and research directions. Future Gen. Comp. Syst. **79**, 849–861 (2018)
2. https://cs.wmich.edu/~elise/courses/cs631/Height-and-Weight-Balanced-Trees.pptx. Accessed 06 Aug. 2019
3. Stojmenovic, I., Wen, S.: The fog computing paradigm: scenarios and security issues. In: 2014 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 1–8. IEEE (2014)
4. https://media.blackhat.com/bh-us-11/Nakibly/BH_US_11_Nakibly_Owning_the_Routing_Table_WP.pdf. Accessed 07 August 2019
5. Harmon, R.R., Auseklis, N.: Sustainable IT services: assessing the impact of green computing practices. In: 2009 Portland International Conference on Management of Engineering & Technology (PICMET), pp. 1707–1717. IEEE (2009)
6. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, pp. 13–16. ACM (2012)
7. https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-solutions.pdf. Accessed 07 Aug. 2019
8. Agrawal, P., Rao, S.: Energy-aware scheduling of distributed systems using cellular automata. In: 2012 IEEE International Systems Conference (SysCon), pp. 1–6. IEEE (2012)
9. Agrawal, P., Rao, S.: Energy-aware scheduling of distributed systems. IEEE Trans. Autom. Sci. Eng. **11**(4), 1163–1175 (2014)
10. Mitra, A., Kundu, A., Chattopadhyay, M.: Energy efficient task-pull scheduling using equal length cellular automata in distributed computing. In: 2014 Fourth International Conference of Emerging Applications of Information Technology, pp. 40–45. IEEE (2014)
11. Mitra, A., Kundu, A.: Energy efficient CA based page rank validation model: a green approach in cloud. Int. J. Green Comput. (IJGC) **8**(2), 59–76 (2017)
12. Mitra, A.: On investigating energy stability for cellular automata based page rank validation model in green cloud. Int. J. Cloud Appl. Comput. (IJCAC) **9**(4), 66–85 (2019)
13. Saha, S., Mitra, A.: Towards exploration of green computing in energy efficient optimized algorithm for uses in fog computing. In: International Conference on Intelligent Computing and Communication Technologies, pp. 628–636. Springer, Singapore (2019)
14. Yu, R., Xue, G., Zhang, X.: Application provisioning in fog computing-enabled internet-of-things: a network perspective. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications, pp. 783–791. IEEE (2018)
15. Oma, R., Nakamura, S., Duolikun, D., Enokido, T., Takizawa, M.: An energy-efficient model for fog computing in the internet of things (IoT). Internet Things **1**, 14–26 (2018)
16. Miyandoab, F.D., Ferreira, J.C., Tavares, V.M.G.: Analysis and evaluation of an energy-efficient routing protocol for WSNs combining source routing and minimum cost forwarding. J. Mobile Multimedia **14**(4), 469–504 (2018)

A Systematic Overview of Fault Tolerance in Cloud Computing

Binod Kumar Pattanayak, Nagarjuna Hota, and Jyoti Prakash Mishra

Abstract Emergence of cloud computing technology has created a revolutionary change in the business world where in a central repository of services and resources can be made available to enterprises on demand. The growing popularity as well as demand of cloud services has made this modern technology more challenging in seamless functionality of it. In order to provide the seamless functionality to cloud computing environment, it must be capable of functioning in an error free manner, and moreover, it must be fault tolerant as well. A set of issues and challenges are associated with provisioning of fault tolerance in cloud computing environment, and at the same time, various strategies have been innovated by researchers and scientists in the said filed to make cloud services fault tolerant. In this paper, we have comprehensively outlined the issues and challenges related to fault tolerance of cloud computing environment along with an elaboration of different fault tolerance provisioning techniques.

Keywords Cloud computing · Fault tolerance · Proactive · Reactive

1 Introduction

Cloud computing is an emerging and most exciting technology as it has a large computing capacity, storage space, applications, and other IT resources to the end user through a cloud service platform via Internet or cloud with a less and affordable

B. K. Pattanayak · N. Hota (✉)
Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India
e-mail: nagarjunahota831@gmail.com

B. K. Pattanayak
e-mail: binodpattanayak@soa.ac.in

J. P. Mishra
Computer Science and Engineering Department, Gandhi Institute for Education and Technology, Baniatangi, Bhubaneswar, Odisha, India
e-mail: jpmishra@gietbbsr.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_2

cost [1]. In cloud computing, the term cloud refers to a metaphor of unstructured resources that are classified into three service domains: (a) application or software, (b) platform, and (c) infrastructure [2]. As cloud represents platform in many fields and that can be distributed in many aspects, it provides all kinds of software, hardware, and network resources to execute customers' specific requirements as well as to manage and store the user data. Therefore, fault tolerance is a necessary and an important property to achieve high performance levels of all attributes and services of cloud computing, in which the vendors of cloud computing should utilize the performance-related issues such as reliability, dependability, availability, throughput, and Quality of Service(QoS). Fault tolerance is defined as a system which may be enable to continue its execution properly in any condition if there is a failure of some of its components and it also has the mechanism that helps in enhancing the performances of cloud by providing different services and platforms to the users as per their demand [1, 3, 4]. There are so many techniques of fault tolerance that are used to create the fault tolerance capability in the cloud which may be categorized into three major types such as redundancy techniques, tolerance policies, and the last one is the load balancing of fault tolerance. In the tolerance policies, two types of fault tolerance schemes are used such as proactive and reactive [2]. For increasing the reliability of fault tolerance technique, we aim at proposing various mechanisms and algorithms. This work represents a comprehensive overview of the fault tolerance-related issues in cloud computing and provides the uses of different fault tolerance methodologies, components, and schemes to increase system performance for cloud computing.

According to the U.S. National Institute of Standards and Technology (NIST) definition: "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be quickly provisioned and released with least management effort or service provider interaction" [5]. Cloud computing offers various resources in the form of services to the end users on demand basis. It enables businesses and users to use applications without installing them on physical machines and allows access to required resources over the Internet. In a layman term, we can define cloud computing as a way of storing and retrieving user's data and programs from a remote server over the Internet instead of using hard drive of computer or any local server. It can also be called as Internet-based computing on cloud servers [6, 7].

The rest of the paper is organized as follows. Section 2 covers the basic concepts related to cloud computing. Section 3 elaborates the various characteristics of cloud computing. Different deployment models of cloud are discussed in Sect. 4. Different types of cloud service models are detailed in Sect. 5. A brief overview of fault tolerance is included in Sect. 6. Fault tolerance in cloud computing is described in Sects. 7 and 8 concludes the paper.

2 Basic Components of Cloud Computing

Cloud computing consists of various components. Each component is having an aim of specifying various roles which can be classified into the following types [8]:

Clients: These are generally the computing systems used by end users to manage data storage and access from cloud servers, which can be a desktops, mobile phones, laptops, iPADs, etc.

Data centers: These are the collections of nodes or various servers where cloud services are taken placed. Using virtualization technique, numbers of virtual servers are created in the data centers which allow the users to share a single instance of resource.

Distributed servers: There are different types of servers which are placed at various locations on earth provided they are equipped with highly scalable machines to keep the server cool. It is capable of better broad network access, polling of resources, reliability, and security to the customers that are using services of cloud.

3 Characteristics of Cloud Computing

Cloud computing has the following five key characteristics which are as follows [8, 9]:

1. ***On-Demand Self Services***: It allows users to directly log on to the Web sites to access cloud sever without any third party human interactions.
2. ***Broad Network Access***: All the resources of cloud can be made available in huge amount over the network. The client may be easily accessed from cloud platforms at any time from any location.
3. ***Multi Tenancy and Resource Pooling***: Resources of cloud computing are means to design to support multi-tenant models and by resource pooling numbers of users can get cloud service from the same shared resource pool.
4. ***Rapid Elasticity and Scalability***: It provides all the capabilities or the resources when the customer requires them and removes when they do not require it. With the scalability of cloud computing, there is a less cost expenditure on the end user side.
5. ***Measured service***: The utilization of cloud resources can be minimized by charge-per-use schemes. All these are get monitored by the cloud service providers.

4 Cloud Deployment Models

The most commonly used models for deployment in cloud environments are as follows [1, 8]:

1. **Public Cloud**: The public cloud permits the general public to access the systems and services offered by an enterprise provider.
2. **Private Cloud**: The private cloud is used within a particular organization. This model ensures high application and data security and privacy.
3. **Community Cloud**: This model is used by various organizations simultaneously.
4. **Hybrid Cloud**: This model is an alliance of both public cloud and private cloud.

5 Cloud Service Models

Cloud computing is based on service models which are also known as the reference models. Cloud service models are of following types [1]:

1. Software-as-a Service (SaaS)
2. Platform-as-a-Service (PaaS)
3. Infrastructure-as-a-Service (IaaS).

1. **Software-as-a-Service (SaaS)**: Through this model, the application softwares are presented by cloud service providers in the form of services to the consumer/end users.
2. **Platform-as-a-Service (Paas)**: This model provides a platform to develop, run, test, and manage applications in the cloud.
3. **Infrastructure-as-a-Service (IaaS)**: The IaaS model provides facility to access to some primary resources, i.e., physical machines, storage, networks, servers, virtual machines on the cloud, etc. (Fig. 1).

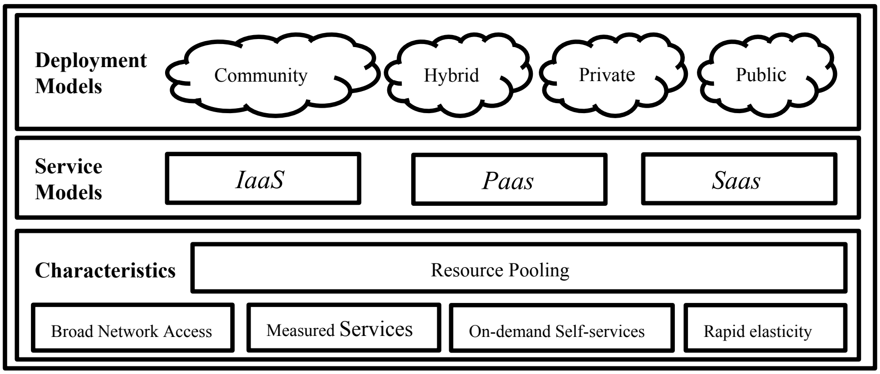


Fig. 1 Different models of cloud computing [14]

6 Fault Tolerance

Fault tolerance is a technique that enables a system for continuing its operation or execution properly if there is an occurrence of the failure of some of its components. It also helps in enhancing the performances of cloud by providing different services and platforms to the users as per their demand. Failures in a system are results by errors which are causes due to faults [1, 2].

Fault tolerance mainly falls into two categories as detailed below:

1. Software fault tolerance
2. Hardware fault tolerance.

Software Fault Tolerance: Here, the main aim of the fault tolerance is to detect and recover the fault of software which is occurring or already occurred in some of the software or hardware component of the cloud server. Major software fault tolerance approaches are recovery blocks, N-version programming, and self-checking software, etc. [10].

Hardware Fault Tolerance: The objective of this is to make the computing machine to work continuously without any disturbances if there is some fault occurs in some of its hardware components [10].

7 Fault Tolerance in Cloud Computing

Fault tolerance in cloud computing is about to develop a blueprint for the system to continue its execution of ongoing task if there is few parts are down or unavailable. Fault tolerance is the capabilities of cloud server nodes that can perform continuous operation even in the occurrence of some faults or failures. It should have the capacity for the fast repairing or replacement of faulty nodes in case of any failure arises. These faults may occur due to some hardware or software components [1–3].

7.1 Fault Tolerance Approaches

Fault tolerance approach plays a vital and important role in detecting and handling of faults that occurs in system. Fault tolerance is a very crucial part in cloud computing platform, though it gives assurance to performance, reliability, and availability of the application software. To achieve high robustness in the cloud systems, it is very essential to access and handle the failure with simple ways and efficiently. The fault tolerance approaches are generally divided into two broad categories such as [1–3, 8, 11]:

1. Reactive fault tolerance
2. Proactive fault tolerance.

Reactive Fault Tolerance: These approaches are basically used to handle any type of faults that occur in a system after they have actually found in the system. It provides robustness or reliability to a cloud system [1, 3] (Table 1).

Proactive Fault Tolerance: These approaches are used to handle the faults in a system after they have actually shown up by the system. Here, the state of the system is continuously monitored to avoid faults and errors [1, 3] (Table 2).

Table 1 Reactive fault tolerance techniques

Reactive fault tolerance techniques	Description
Checkpoint [1, 3]	Basically, it is used to save state of the systems in certain time period. If there is a continuous task failure occurs, the task can continue its work from the checkpoint instead of restarted from beginning
Job migration [1, 3, 15]	If an executing task cannot be completed on a particular node due to some of the reason of failure, it can be migrated to some other node
Replication [1, 3]	It is used to create different numbers of copies of a task and stores it may be on different node so that a task can continue its execution if there is some failure occurs
S-guard [1]	It is based on rollback and recovery process
Retry [8, 15]	Here, the task is repeatedly executed until it succeeds
Task resubmission [8]	Here, the task that gets failed again is submitted to the same node or any other machine for its execution
Rescue workflow [8]	System is allowed to continue its work

Table 2 Proactive fault tolerance techniques

Proactive fault tolerance techniques	Description
Self-healing [1, 3]	It is based on divide and conquer technique. In this method, first the task is divided into number of chunks, performs its execution and again merged back without any error
Software rejuvenation [8, 15]	Here, the system always reboots and starts its job from the beginning
Pre-emptive migration [1, 8, 15]	In this method, the task is continuing its work depending upon a mechanism of feedback loop flow
Load balancing [1]	By using this method, the loads of a node can be balanced among with the other nodes or with the CPU by transferring loads from a heavily loaded node to an under loaded node

7.2 Fault Tolerance Parameters

Some parameters are used to identify the efficiency of the system performance on cloud which are as follows [8, 12, 13]:

Throughput

It can be defined as the number of tasks that has been executed successfully per unit time. Any computing system must have a high throughput.

Scalability

Higher the number of computing nodes in the distributed environment of cloud at any point of time should not affect the fault tolerance capacity of the algorithm.

Response time

Time taken by the system to respond to an user's request is known as response time. System having a lesser response time is good for client.

Usability

The higher is the usability, the easier it is for the user to achieve goals. The application is said to be usable if it can be used in several extent to support the user in order to provide good services with little or no modification.

Availability

It is the context of the computing machine to be executable at a proportion of time. A system with high availability is trusted to provide better service to the client.

Adaptability

It can be defined as the ability of the computing system to accept the current computing environment and make some changes to its modules according to its need.

Related overhead

The overhead in the fault tolerance can be distinguished during the execution of an algorithm. The occurrence of overhead is due to the occurrence of overhead of execution of tasks in the processors and the communication between them.

Effect on cost

Here, the cost can be only determined during the purchasing of cloud services.

8 Conclusion

Computing in the cloud provides various features like scalability, elasticity, high availability, and many more. The cloud computing model has changed the IT industry as it brings several benefits to individuals, researchers, organizations, and even countries. Despite providing numerous advantages, the cloud system is still susceptible to failures. Failures are unavoidable in cloud computing. Various fault tolerance approaches are taken into consideration to implement for handling and managing faults and failures effectively and efficiently on the cloud computing platform with prevent as well as tolerate the system failures that may be happened due to either hardware of software failure. In this paper, the main motivation is to implement appropriate fault tolerance approaches on the cloud computing platform to enhance fault tolerance capacity in the cloud computing environments and to achieve recovery of failure with high reliability and availability.

References

1. Kumari, P., Kaur, P.: A survey of fault tolerance in cloud computing. J. King Saud Univ. Comput. Inf. Sci. doi https://doi.org/10.1016/j.jksuci.2018.09.021
2. Cheraghlou, M.N., Khadem-Zadeh, A., Haghparast, M.: A survey of fault tolerance architecture in cloud computing. J. Netw. Comput. Appl. **61**, 81–92 (2016)
3. Hasan, M., Goraya, M.S.: Fault tolerance in cloud computing environment: a systematic survey. Comput. Ind. **99**, 156–172 (2018)
4. Kumar, A., Malhotra, D.: Study of various reactive fault tolerance techniques in cloud computing. Int. J. Comput. Sci. Eng. **06**(Special Issue 05), 54–58 (2018)
5. Mell, P., Grance, T.: The NIST definition of cloud computing recommendations of the national institute of standards and technology. NIST Spec. Publ. **145**, 7 (2011)
6. https://www.geeksforgeeks.org/cloud-computing/
7. https://www.tutorialspoint.com/cloud_computing/index.htm
8. Amin, Z., Singh, H., Sethi, N.: Review on fault tolerance techniques in cloud computing. Int. J. Comput. Appl. **116**(18) (2015)
9. https://www.controleng.com/articles/five-characteristics-of-cloud-computing/
10. Sivagami, V.M., Easwara Kumar, K.: Survey on fault tolerance techniques in cloud computing environment. Int. J. Sci. Eng. Appl. Sci. **1**(9), 419–425 (2015)
11. Liu, J., Wang, S., Zhou, A., Kumar, S.A., Yang, F., Buyya, R.: Using proactive fault-tolerance approach to enhance cloud service reliability. IEEE Trans. Cloud Comput. **6**(4), 1191–1202 (2016)
12. Prathiba, S., Sowvarnica, S.: Survey of failures and fault tolerance in cloud. In: International Conference on Computing and Communications Technologies (ICCCT), pp. 169–172. IEEE (2017)
13. Seyyed, M.H., Mostafa, G.A.: Fault-tolerance techniques in cloud storage: a survey. Int. J. Database Theory Appl. **8**(4), 183–190 (2015)
14. Alkasem, A., Liu, H.: A survey of fault-tolerance in cloud computing: concepts and practice. Res. J. Appl. Sci. Eng. Technol. **11**(12), 1365–1377 (2015)
15. Alshayeji, M., Al-Rousan, M., Yossef, E., Ellethy, H.: A study on fault tolerance mechanisms in cloud computing. Int. J. Comput. Electr. Eng. **10**, 574–538 (2018)
16. Bilal, K., Khalid, O., Malik, S.U.R., Khan, M.U.S., Khan, S.U., Zomaya, A.Y.: Fault tolerance in the cloud. In: Encyclopedia of Cloud Computing (pp. 291–300). Chichester: Wiley (2016)

17. Tebaa, M., Hajji, S.E.: From single to multi-clouds computing privacy and fault tolerance. IERI Procedia **10**, 112–118 (2014)
18. Roveicy, M.R., Bidgoli, A.M.: Comparative analysis of fault tolerance techniques in cloud computing: a case of Armangerayan Co. Int. J. Eng. Technol. **7**(3.14), 432–436 (2018)

An EPIC Mechanism for Advertising the Tourist Spots in Smart Cities

Vikash Kumar Singh, Anil Bikash Chowdhary, and Sajal Mukhopadhyay

Abstract In this paper, we have investigated the problem of advertising the tourist spots present in the multiple cities (mainly *smart cities*) through the involvement of agents (*travel agents*, *local residents*, etc.), in *strategic* setting. For advertising the tourist spots, each agent (as *task executor*) can charge a value (maximum willingness for performing the task), called the *valuation*. In our set-up, the *valuation* is *private* and is only known to him. Given such scenario, the objective is to select a subset of task executors such that all the tourist spots get advertised along with the constraint of maximizing the social welfare. For this purpose, we have designed an *ex-post incentive compatible* (EPIC) mechanism based on the descending price auction.

Keywords Smart cities · Tourist spots · EPIC mechanism

1 Introduction

Around the globe, the cities are getting densely populated and are growing day by day. According to one of the reports of United Nations (*published in May 2018*), around 55% of the world's population lives in cities or urban areas and this figure is expected to increase to approximately 68% by the year 2050 [1]. Hence, with the rapid expansion in the cities, there is an alarming need to manage *energy resources*,

V. K. Singh (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: vikashsingh@soa.ac.in

A. B. Chowdhary
Department of Computer Applications, Techno India University, Kolkata, West Bengal 700091, India
e-mail: abchaudhuri007@gmail.com

S. Mukhopadhyay
Department of Computer Science and Information Technology, National Institute of Technology Durgapur, Durgapur, West Bengal 713209, India
e-mail: sajal@cse.nitdgp.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*, Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_3

transport facilities, *food and shelter*, and *available spaces* in an effective and efficient way. In order to meet the above-raised challenges, the modern technology and the Internet will be playing a crucial role up to some extent [2]. Another idea that is in current practice is that many cities are involving the local residents in the decision-making. For example, the residents by their voluntary participation provide the data about the presence of various toxic materials in their respective geographical region [3]. In past, some works have been done that studies the involvement of local residents in decision-making in the travels and tourism sector [4–6], but that in *non-strategic* setting.

The above-presented statistics and discussion strongly motivated us to explore one of the directions in travel and tourism sector where there is an active participation of agents in decision-making, in *strategic* setting. By *strategic setting*, we mean that the participating agents (in our case the *travel agents*, *local residents*, etc.) are *rational* and *intelligent*. Informally, *rationality* of an agent means that the agent chooses the strategies so as to maximize a well-defined individualistic pay-off, while *intelligence* means that agents are fully equipped to compute their best strategies. As the participating agents are *strategic* in nature, the problem is modelled by utilizing the robust concept of *mechanism design with money*.[1]

In this paper, the problem that we have considered is the municipalities from multiple cities provide the tourist spots that are to be advertised, to the *platform*. On receiving the tourist spots, the *platform* projects the task of advertising the tourist spots to the outside world where the agents (as *task executors*) are present. Now, on seeing this, the task executors submit the information about the number of tourist spots they want to advertise along with the value they will charge (or *bid*) called the *valuation*, for advertising the tourist spot(s). It is to be noted that the *valuation* of each agent is *private* information and is only known to him. Given this set-up, the goal is to hire a set of task executors for advertising the tourist spots with the constraint of maximizing the *social welfare* (see Definition 1). For this purpose, in this paper, an *ex-post incentive compatible* (EPIC) mechanism (see Definition 2) motivated by Roughgarden [7] and Nisan et al. [8] is proposed.

The main contribution of this paper is:

- The problem of advertising the tourist spots of the multiple cities that involve strategic agents in decision-making is modelled using *mechanism design with money*.
- An *ex-post incentive compatible* (EPIC) mechanism based on the descending price auction is proposed, namely *posted price mechanism for the tourist spots advertisement problem* (PPM-TSAP) motivated by Roughgarden [7] and Nisan et al. [8].

The remainder of this paper is organized as follows. In Sect. 2, the literature work in the travel and tourism is covered. We describe our proposed system model in detailed manner in Sect. 3. In Sect. 4, the proposed mechanism is presented and discussed. We conclude this paper and provide the future direction in Sect. 5.

[1] By *mechanism design with money*, we mean that the agents demand money for their imparted services.

2 Related Works

In past, several works have been done that mainly talk about the involvement of residents in the decision-making for various aspects in tourism. Nunkoo et al. [9] endeavoured 140 papers that have been published between 1984 and 2010 that mainly covers the direction of residents' attitude towards the tourism. The studies about the community's attitudes to the tourism have increased rapidly over recent decades, reflecting the importance of residents' involvement in tourism for the sustainable development of the sector [4, 5]. It has been seen that researchers support the fact that the participation and cooperation of the residents are crucial factors for sustainable tourism development [6]. In order to have some quick review about the status of research on this topic, we recommend readers to go through [9, 10]. In [11], there discussion circumvents around the impact of residents' personality on the overall tourism development. Research has shown that [12] if residents are happier with their life in general, they are also more willing to interact with tourists. It has been found that the breakthrough work by Arnstein [13] has formed the basis of works in the direction of residents' participation in a various decision-making in the projects of public interest. In [14, 15], the underlying principle is discussed that plays an important role in dragging large number of residents in decision- and policy-making.

3 System Model and Problem Formulation

In this section, we present the formal statement of our problem. The municipalities of the multiple cities want to advertise the set of n tourist spots given as $\mathbb{T} = \{\mathbb{T}_1, \mathbb{T}_2, \ldots, \mathbb{T}_n\}$. In this, each of the municipalities is having an *infinite* budget (he can expend money based on the demand). On the other hand, we have multiple task executors say m that are interested to advertise the tourist spots. The set of m task executors is represented as $\mathbb{E} = \{\mathbb{E}_1, \mathbb{E}_2, \ldots, \mathbb{E}_m\}$. Each task executor $\mathbb{E}_i$ has a *marginal valuation*, which is *private* and is denoted by $v_i(j)$ for jth tourist spot.[2] Here, $v_i(j)$ is the valuation of task executor $\mathbb{E}_i$ for the jth tourist spot given that $\mathbb{E}_i$ already have $j-1$ tourist spots. So, the total valuation of the ith task executor for t tourist spots is $v_i(t) = \sum_{k=1}^{t} v_i(k)$. In our set-up, all the task executors possess a *downward-sloping* valuations.[3] Let the price slapped on each of the tourist spots be $\tilde{\boldsymbol{p}}$. Given the price $\tilde{\boldsymbol{p}}$, the utility of the task executor $\mathbb{E}_i$ for advertising ℓ tourist spots is represented as $u_i(\tilde{\boldsymbol{p}})$ and is defined as:

$$u_i(\tilde{\boldsymbol{p}}) = \begin{cases} \ell \cdot \tilde{\boldsymbol{p}} - v_i(\ell), & \text{if } i\text{th task executor advertise } \ell \text{tourist spots,} \\ 0, & \text{Otherwise.} \end{cases} \tag{1}$$

[2]For any task executor $\mathbb{E}_i$, by *private* marginal valuation we mean that the marginal valuation will be known to the task executor $\mathbb{E}_i$ only and not known to others.

[3]For any task executor $\mathbb{E}_i$, by *downward-sloping* valuation we mean that $v_i(1) \geq v_i(2) \geq v_i(3) \geq \ldots \geq v_i(n)$.

One of the important components of the task executors that is perceived in our proposed model is the *supply* of the task executors at price $\tilde{\boldsymbol{p}}$. We can say that the *supply* of any task executor $\mathbb{E}_i$ at a given price $\tilde{\boldsymbol{p}}$ is the set of tourist spots that maximizes the value $u_i(\tilde{\boldsymbol{p}})$ and is given as $\mathbb{S}_i(\tilde{\boldsymbol{p}}) = \arg \max_{x \in \{0,\ldots,m\}} u_i(\tilde{\boldsymbol{p}})$. The total supply at price $\tilde{\boldsymbol{p}}$ is the sum of the supply of all the m task executors and is given as $\mathbb{S} = \sum_{j=1}^{m} \mathbb{S}_j(\tilde{\boldsymbol{p}})$. The payment made to the ith task executor for tourist spot $\mathbb{T}_j$ is denoted by $p_i(j)$. For all the n tourist spots, the payment vector of a task executor $\mathbb{E}_i$ is given as $\mathbb{P}_i = \{p_i(1), p_i(2), \ldots, p_i(n)\}$. The payment vector of all the task executors is given as $\mathbb{P} = \{\mathbb{P}_1, \mathbb{P}_2, \ldots, \mathbb{P}_m\}$. Given the above-discussed set-up, the goal is to select the set of task executors such that the tourist spots get advertised along with the constraint of maximizing the social welfare.

Definition 1 (*Social Welfare* (*SW*)) It is the sum of the valuations of the task executors for the tourist spots. Mathematically, we have SW = $\sum_{i=1}^{m} \sum_{j=1}^{n} v_i(j)$.

Definition 2 (*Ex-post Incentive Compatible* (*EPIC*) [8]) A mechanism is said to be an EPIC if sincere bidding is an ex-post Nash equilibrium that guarantees all agents non-negative utility.

4 Tourist Spots Advertisement Problem: Posted Price Mechanism

In this section, we have proposed an EPIC mechanism, namely PPM-TSAP, motivated by Roughgarden [7] and Nisan et al. [8]. The PPM-TSAP consists of the *allocation rule* and *pricing rule*.

4.1 Outline of PPM-TSAP

In this section, the underlying idea of PPM-TSAP is presented. The PPM-TSAP is based on *descending price auction* that stops once the sufficient number of task executors is available to advertise the tourist spots.

PPM-TSAP

Allocation Rule:

1. Initially, set the price $\tilde{\boldsymbol{p}}$ as a very high value.
2. **While** (True)
   (a) From each task executor i, ask the question that how many number of tourist spots he/she (henceforth he) wants to advertise at the given price $\tilde{\boldsymbol{p}}$ i.e. $\mathbb{S}_{\mathrm{i}}(\tilde{\boldsymbol{p}})$, value.
   (b) If $\sum_{i=1}^{m} \mathbb{S}_i(\tilde{\boldsymbol{p}}) \leq n$, then go to step 3.
   (c) Else, decrement $\tilde{\boldsymbol{p}}$ by ϵ value (ϵ is a small constant that is fixed apriori.).
3. Let $\tilde{\boldsymbol{p}}$ be the price at which the **while** loop terminates. For each task executor $\mathbb{E}_i$, the allocation resulted is given as: $\boldsymbol{\alpha}_i = [\mathbb{S}_i(\tilde{\boldsymbol{p}}),\ \mathbb{S}_i(\tilde{\boldsymbol{p}} + \epsilon)]$ such that $\sum_{i=1}^{m} \boldsymbol{\alpha}_i = n$.

Pricing Rule: The payment of any ith task executor for the jth tourist spot is:

$$p_i(j) = -\epsilon + \max_{t \in \mathbb{Z}^+} \left\{ \epsilon t \;\middle|\; \sum_{k \neq i} \mathbb{S}_k(\epsilon t) \leq n - j,\ \forall j = 1,\ \ldots,\ n \right\}$$

4.2 Detailing of PPM-TSAP

The PPM-TSAP consists of PPM-TSAP *allocation rule* (depicted in Algorithm 1) and PPM-TSAP *pricing rule* (depicted in Algorithm 2). In line 1 of the Algorithm 1, the price $\tilde{\boldsymbol{p}}$ is set to very high value (say ∞). The *for* loop in line 2–4 sets the supply of each task executor as n. In line 5, the total supply of the task executors is calculated at price $\tilde{\boldsymbol{p}}$. Now, with the decrease in price $\tilde{\boldsymbol{p}}$ the actual supply calculation is done by the *while* loop in line 6–12. In lines 13–17, the allocation is done to the task executors at price $\tilde{\boldsymbol{p}}$. In line 15, the track of number of tourist spots allocated to each of the task executors $\mathbb{E}_i$ is kept. The $\boldsymbol{\beta}$ data structure holds the task executors. Lines 18–26 capture the case when at price $\tilde{\boldsymbol{p}}$ the total supply is less than the available tourist spots to be advertised. So, using *while* loop in lines 19–25 the task executors are allocated randomly to the remaining tourist spots but this time at the price $\tilde{\boldsymbol{p}} + \epsilon$. Finally, in line 27 the allocation vector is returned. Next, the *pricing rule* of PPM-TSAP is depicted in Algorithm 2. In lines 1–7 of Algorithm 2, the payment of each of the task executors is calculated. Once the payment of each of the task executors is calculated, the payment vector $\mathbb{P}$ is returned in line 8.

Algorithm 1: PPM-TSAP Allocation rule ($\mathbb{T}$, $\mathbb{E}$, ϵ)

Output: $\boldsymbol{\alpha} \leftarrow \phi$

$\tilde{\boldsymbol{p}} \leftarrow \infty$
foreach $i = 1$ *to* m **do**
  $\mathbb{S}_i(\tilde{\boldsymbol{p}}) = n$
end
$\mathbb{S} = \sum_{j=1}^{m} \mathbb{S}_j(\tilde{\boldsymbol{p}})$ `// Total supply at price` $\tilde{\boldsymbol{p}}$.
while $\mathbb{S} > n$ **do**
  $\tilde{\boldsymbol{p}} \leftarrow \tilde{\boldsymbol{p}} - \epsilon$
  foreach $\mathbb{E}_i \in \mathbb{E}$ **do**
    $\mathbb{S}_i(\tilde{\boldsymbol{p}}) = \arg\max_{x \in \{0,\ldots,n\}} u_i(\tilde{\boldsymbol{p}})$
  end
  $\mathbb{S} = \sum_{j=1}^{m} \mathbb{S}_j(\tilde{\boldsymbol{p}})$
end
foreach $\mathbb{E}_i \in \mathbb{E}$ **do**
  $\boldsymbol{\alpha}_i = \mathbb{S}_i(\tilde{\boldsymbol{p}})$ `//` $\mathbb{S}_i(\tilde{\boldsymbol{p}})$ `no. of tourist spots is allocated to` $\mathbb{E}_i$.
  $\boldsymbol{\alpha} = \boldsymbol{\alpha} \cup \{\boldsymbol{\alpha}_i\}$
  $\boldsymbol{\beta} = \boldsymbol{\beta} \cup \{\mathbb{E}_i\}$
end
if $n - \mathbb{S} > 0$ **then**
  while $n - \mathbb{S} \neq 0$ **do**
    $f_k \leftarrow random\ (\mathbb{E} \setminus \boldsymbol{\beta},\ \tilde{\boldsymbol{p}} + \epsilon)$ `// Randomly selects one of the task executors that was interested to advertise the tourist spot at` $\tilde{\boldsymbol{p}} + \epsilon$.
    $\boldsymbol{\alpha}_k \leftarrow min\{\mathbb{S}_k(\tilde{\boldsymbol{p}} + \epsilon),\ n - \mathbb{S}\}$
    $\mathbb{S} = \mathbb{S} + \boldsymbol{\alpha}_k$
    $\boldsymbol{\alpha} = \boldsymbol{\alpha} \cup \{\boldsymbol{\alpha}_k\}$
    $\boldsymbol{\beta} = \boldsymbol{\beta} \cup \{\mathbb{E}_k\}$
  end
end
return $\boldsymbol{\alpha}$

Algorithm 2: PPM-TSAP Pricing rule ($\mathbb{T}$, $\mathbb{E}$, ϵ)

foreach $i = 1$ *to* m **do**
  foreach $j = 1$ *to* n **do**
    $p_i(j) = -\epsilon + \max\limits_{t \in \mathbb{Z}^+} \left\{ \epsilon t \;\middle|\; \sum\limits_{k \neq i} \mathbb{S}_k(\epsilon t) \leq n - j \right\}$
    $\mathbb{P}_i \leftarrow \mathbb{P}_i \cup \{p_i(j)\}$
  end
  $\mathbb{P} \leftarrow \mathbb{P} \cup \{\mathbb{P}_i\}$
end
return $\mathbb{P}$

4.3 Illustrative Example

The set-up consists of three tourist spots and five task executors. The marginal valuations of task executors are depicted in Fig. 1a. The supply on several prices is shown in Fig. 1b. At price 17, the supply of each task executor is 3 tourist spots. The total supply is 15.
Now, the price is decremented by 7, and at price 10, the supply of each task executor is 3 tourist spots. So, the total supply is still 15. Next, the price is decremented by 7 and current price becomes 3. At price 3, the mechanism terminates as the total supply equals the number of tourist spots. At this price, task executors $\mathbb{E}_3$, $\mathbb{E}_4$ and $\mathbb{E}_5$ clinch one tourist spot each. The payment made to $\mathbb{E}_3$, $\mathbb{E}_4$ and $\mathbb{E}_5$ is 3, 3 and 3, respectively.

Tourist Spots	Task Executors				
	$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_3$	$\mathbb{E}_4$	$\mathbb{E}_5$
First	5	4	7	5	10
Second	4	4	5	4	4
Third	4	2	3	3	4

(a) Initial Set-up

Price	Task Executors				
	$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_3$	$\mathbb{E}_4$	$\mathbb{E}_5$
17	3	3	3	3	3
10	3	3	3	3	3
03	0	0	1	1	1

(b) Price and Demand of Task Executors

Fig. 1 Detailed illustration of PPM-TSAP (with $\epsilon = 7$)

4.4 Running Time of PPM-TSAP

In Algorithm 1, line 1 is bounded from above by $O(1)$. The *for* loop in lines 2–4 will take $O(m)$ time. Line 5 is bounded above by $O(1)$. For each iteration of *while* loop in lines 6–12, the *for* loop in lines 8–10 will take $O(mn)$ time. Now, as each time ϵ is subtracted from the price $\tilde{\boldsymbol{p}}$, so in the worst case lines 7–11 of the *while* loop will execute for $\frac{\tilde{p}}{\epsilon}$. So, the overall running time of the *while* loop in lines 6–12 will be $O(mn)$ as $\frac{\tilde{p}}{\epsilon}$ is constant. The *for* loop in lines 13–17 is bounded from above by $O(m)$. For each iteration of *while* loop in lines 19–25, lines 20–24 are bounded from above by $O(m)$. In the worst case, the *while* loop in lines 19–25 will iterate for n times. So, lines 18–26 are bounded from above by $O(mn)$. So, the running time of Algorithm 1 is $O(1) + O(m) + O(mn) + O(mn) = O(mn)$. In Algorithm 2, for each iteration of *for* loop in lines 1–7, the inner *for* loop is bounded from above by $O(mn)$. So, for each iteration of *for* loop in lines 2–5, the price calculation takes $O(m)$ time. So, lines 1–7 will take $O(m^2n)$ time in worst case. So, the running time of PPM-TSAP *pricing rule* (Algorithm 2) is $O(m^2n)$. So, the running time of PPM-TSAP is $O(mn) + O(m^2n) = O(m^2n)$.

Definition 3 Given a sample space S and an event $\mathscr{A}$ in the sample space S, let $X_{\mathscr{A}} = I\{\mathscr{A}\}$. Then, $E[X_{\mathscr{A}}] = Pr\{\mathscr{A}\}$ [16].

Lemma 1 *The expected number of tourist spots any task executor $\mathbb{E}_k$ is advertising is given as $p \cdot z_k$, where z_k is the number of tourist spots he is interested to advertise and p is the probability with which the task executor $\mathbb{E}_k$ is considered for a tourist spot. More formally, $E[X_k] = p \cdot z_k$, where X_k is the random variable measuring the number of tourist spots $\mathbb{E}_k$ is advertising out of z_k.*

Proof Fix a task executor $\mathbb{E}_k$. Our aim is to calculate the expected number of tourist spots $\mathbb{E}_k$ is advertising out of z_k. The sample space for $\mathbb{E}_k$ for any given tourist spot $\mathbb{T}_l$ is $S = \{\mathbb{E}_k$ is advertising tourist spot $\mathbb{T}_l$, $\mathbb{E}_k$ is not advertising tourist spot $\mathbb{T}_l\}$. Now, we have the $Pr\{\mathbb{E}_k$ is advertising tourist spot $\mathbb{T}_l\} = p$ and $Pr\{\mathbb{E}_k$ is not advertising tourist spot $\mathbb{T}_l\} = 1 - p$. We can define an indicator random variable X_{kl}, associated with $\{\mathbb{E}_k$ is advertising tourist spot $\mathbb{T}_l\}$, which is event $\mathscr{A}$. So, we can write

$$X_{kl} = I\{\mathscr{A}\} \tag{2}$$

$$= \begin{cases} 1, & \text{if} \mathscr{A} \text{occurs,} \\ 0, & \text{otherwise.} \end{cases}$$

The expected number of tourist spots $\mathbb{E}_k$ is advertising is simply the expected value of our indicator random variable X_{kl}, so taking expectation both sides of Eq. (2), we get:

$$E[X_{kl}] = E[I\{\mathscr{A}\}]$$

Following Definition 3, we have:

$$E[X_{kl}] = 1 \cdot Pr\{X_{kl} = 1\} + 0 \cdot Pr\{X_{kl} = 0\}$$
$$= 1 \cdot p + 0 \cdot (1 - p)$$
$$E[X_{kl}] = p \tag{3}$$

Thus, the expected number of times $\mathbb{E}_k$ is advertising tourist spot $\mathbb{T}_l$ is p. Now, let us consider the random variable that we are interested in and it is given as $X_k = \sum_{l=1}^{z_k} X_{kl}$. Here, we can compute $E[X_k]$ by taking expectation both sides:

$$E[X_k] = E\left[\sum_{l=1}^{z_k} X_{kl}\right]$$

By linearity of expectation and Eq. (3), we have

$$E[X_k] = \sum_{l=1}^{z_k} E[X_{kl}] = \sum_{l=1}^{z_k} p = p \cdot z_k$$

If we take $p = \frac{1}{2}$, then it can be seen that the value of $E[X_k]$ will be boiling down to $\frac{z_k}{2}$. By this, one can infer that any arbitrary task executor $\mathbb{E}_k$ in expectation will be advertising half of the number of tourist spots for which he has shown interest.

5 Conclusion and Future Work

In this paper, we have investigated the problem of advertising the tourist spots of multiple cities with the help of agents, in *strategic* setting. We have designed an EPIC mechanism for hiring a set of task executors in order to advertise the tourist spots with the constraint of maximizing the social welfare. In our future work, we can think of designing a Dominant Strategy Incentive Compatible (DSIC) mechanism for the problem discussed in Sect. 3.

References

1. DESA, U.: 68% of the world population projected to live in urban areas by 2050, says un (2018), https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html
2. Manville, C., Cochrane, G., Cave, J., Millard, J., Pederson, J.K., Thaarup, R.K., Liebe, A., Wissner, M., Massink, R., Kotternik: Mapping European cities in the EU, 2014. EU par-

liament (2018). Available online: http://www.europarl.europa.eu/RegData/etudes/etudes/join/2014/507480/IPOL-ITRE_ET
3. Hasenfratz, D., Saukh, O., Sturzenegger, S., Thiele, L.: Participatory air pollution monitoring using smartphones. In: Mobile Sensing: From Smartphones and Wearables to Big Data. ACM, Beijing, China (2012)
4. Choi, H.S.C., Sirakaya, E.: Measuring residents attitude toward sustainable tourism: development of sustainable tourism attitude scale. J. Travel Res. **43**(4), 380–394 (2005). https://doi.org/10.1177/0047287505274651
5. Diedrich, A., Garca-Buades, E.: Local perceptions of tourism as indicators of destination decline. Tourism Manage. **30**(4), 512–521 (2009) cited By 130
6. Yu, C.P.S., Chancellor, H.C., Cole, S.T.: Measuring residents attitudes toward sustainable tourism: a reexamination of the sustainable tourism attitude scale. J. Travel Res. **50**(1), 57–63 (2011). https://doi.org/10.1177/0047287509353189
7. Roughgarden, T.: CS364B: Frontiers in Mechanism Design (Stanford University course), lecture 4: The Clinching Auction (2014)
8. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, New York (2007)
9. Nunkoo, R., Smith, S.L.J., Ramkissoon, H.: Residents attitudes to tourism: a longitudinal study of 140 articles from 1984 to 2010. J. Sustain. Tourism **21**(1), 5–25 (2013). https://doi.org/10.1080/09669582.2012.673621
10. Deery, M., Jago, L., Fredline, L.: Rethinking social impacts of tourism research: a new research agenda. Tourism Manage. **33**(1), 64 – 73 (2012). http://www.sciencedirect.com/science/article/pii/S0261517711000422
11. Moghavvemi, S., Woosnam, K.M., Paramanathan, T., Musa, G., Hamzah, A.: The effect of residents personality, emotional solidarity, and community commitment on support for tourism development. Tourism Manage. **63**(C), 242–254 (2017). https://ideas.repec.org/a/eee/touman/v63y2017icp242-254.html
12. Woo, E., Kim, H., Uysal, M.: Life satisfaction and support for tourism development. Ann. Tourism Res. **50**, 84–97 (2015). http://www.sciencedirect.com/science/article/pii/S0160738314001406
13. Arnstein, S.R.: A ladder of citizen participation. J. Am. Inst. Plann. **35**(4), 216–224 (1969). https://doi.org/10.1080/01944366908977225
14. Medaglia, R.: eparticipation research: moving characterization forward (2006–2011). Gov. Inf. Q. **29**(3), 346–360 (2012)
15. Martin, N.J., Rice, J.L., Lodhia, S.K.: Sustainable development planning: a case of public participation using online forums. Sustain. Dev. **22**(4), 265–275 (2014). https://onlinelibrary.wiley.com/doi/abs/10.1002/sd.1541
16. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. MIT press (2009)

Data Analysis and Prediction of Terror Component Using Web Services

Shrabanee Swagatika, Bishnupriya Panda, Debabrata Singh, and Alakananda Tripathy

Abstract Analyzing terrorist activities are crucial not only for the sake of safety but also for the growth of any country. But handling and identifying the terror activity is a big challenge nowadays, where Internet-based services already captured the world, which helps them spreading rumors and fake news for terrifying normal citizen as well as motivating them for wrong goals. Here in this paper, we discussed some means for collecting the data from Webs and trying to analysis to find out such kinds of sources of terrifying entities; for doing this, we have used some techniques like case study, usage data, surveys, interviews, focus groups, etc.

Keywords Terror · Intrusion detection system · Receiver operator characteristic · Vector space

1 Introduction

A terrorist gathering is generally a social framework. The gathering is shaped as a bound together entire by fan resolved to dispense nonmilitary personnel and monetary harm on explicit focuses in quest for their fanatic objectives. When there is an expanding feeling of neurosis in regard to terrorists, there is a solid requirement for adjusted, master, and educated examination into this subject. Great research can give integral assets to understanding and direction on what has turned out to be a standout

S. Swagatika (✉) · B. Panda · D. Singh · A. Tripathy
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: shrabaneeswagatika@soa.ac.in

B. Panda
e-mail: bishnupriyapanda@soa.ac.in

D. Singh
e-mail: debabratasingh@soa.ac.in

A. Tripathy
e-mail: alakanandatripathy@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_4

among the most difficult issues of the advanced age, yet great research has frequently been urgently inadequate. Detection of terrorists on the Web can put a check on further attacks because terrorists also use Internet for exchange of information and to hire new members. But learning the behavior of terrorist groups has become way more complex and hence requires new methods and ways to understand the dynamics. Anticipating terrorist gathering conduct is of extraordinary significance, particularly for assault conduct. When such conduct is anticipated, we can screen the elements of the gathering and identify early cautioning indications of the conduct change. The infamous 'Hamburg cell,' that was responsible behind September 11 attacks in USA also used high-speed Internet connections for exchange of information. Therefore, gathering information from the Web has become vital for law enforcement agencies in order to monitor further attacks by the terrorists. Unfortunately, it is quite challenging to check terrorist Web sites as they do not use fixed IP address and URLs [1–3]. Therefore, monitoring all ISPs has become essential in order to check terrorist activities though privacy issues come into picture in this scenario.

The design criteria of the proposed model here include the following step

- Contents of already available terrorist sites and recognized terrorist traffics on the Web are taken as input to train the detection algorithm.
- Sensitivity of detection should be monitored by parameters defined by user.
- This design must detect in time.

2 Background

However shockingly, little research work of logical legitimacy has been directed on the culprits of terrorist violence. The exercises of terrorist gatherings and the idea of their participation have all around been studiously overlooked by social researchers. There are a couple of exemptions to this standard (more often than not including detainees), yet all in all the scholarly teaches have not raced to the investigation of terrorist [4, 5]. Not many distributed endeavors have been made to methodically examine terrorists outside of a jail setting or to consider in a deliberate way the real exercises did as a major aspect of the terrorism battles. For a sensational wonder of such serious enthusiasm to the media and more extensive world, such expanding gaps in the writing are downright staggering [6].

2.1 Intrusion Detection System (IDS)

IDS is a system or programming application that checks a system or frameworks for malevolent movement or strategy infringement using some tools like OutWit Hub, Web Scraper, Spinn3r, FMiner, ParseHub, Octoparse, Table Capture, etc. It issues alarm when such exercises are found. The IDS breaks down various types of data about activities produced from the earth and ascertains the likelihood that they are interruption side effects [7–9]. Such data incorporates, for instance, design data about the present condition of the framework, review data depicting the occasions that happen in the framework (e.g., occasion sign in Windows XP), or system traffic and the earth incorporate a solitary PC or a few PCs associated in a system or the system itself. The IDS constantly monitors the actions in an environment. There are several methods for evaluating IDS such as accuracy, completeness, efficiency, and fault tolerance. True-positive rate (TP) and false-positive rate (FP) come under most widely used measures to evaluate IDS [10].

2.2 Vector Space Model

It is used in information retrieval, information filtering, indexing, etc. Here, text documents are represented as vectors, such as index terms. One of the major problems in this research is to represent the textual content of the Web pages. Therefore, vector space model is used to represent terrorists' interest [11–13]. Here, a document d is represented by an n-dimensional vector $d = (w^1, w^2, \ldots w^n)$, where w^i represents the frequency-dependent weight of term i in document d. The comparability among two archives spoke to as vectors might be estimated by utilizing Euclidian separation or cosine. In this exploration, each site page is taken as record and portrayed as vector.

2.3 Receiver Operator Characteristic (ROC) Curve

ROC curve is a graph that delineates the demonstrative or diagnostic capacity of a binary classifier framework as its threshold value is differed. Here, we use ROC curve for result analysis. It is a plot between true-positive rate and false-positive rate of a diagnostic test. It is an effective means of evaluating performance. This is widely used in radiology and other medical diagnostic tests. Our research uses this for comparison with command-based intrusion detection system.

3 Content-Based Detection of Terror-Related Activity

3.1 Detection Environment

The research involves Web page details browsed by terrorists or people involved with them or their supporters as input to the detection system. Images, music, video clips, and other complex data are excluded in this process. By taking into consideration the textual content of the Web sites visited by terrorists and their helpers, a 'Typical terrorist behavior' is obtained. Each user under supervision is distinguished as a 'user's computer' having a particular IP address. Hence, the detected IP can be used to find the computer's location and also the terrorist if it is still logged in.

Two modes of operation are associated with the recommended methodology

3.1.1 Learning Typical Terrorist Behavior

Here, the input content is downloaded from a collection of Web sites and represented as vectors in vector space model. It also involves methods/approaches of unsupervised clustering. Here, the IP addresses of the Web pages are not taken into account.

3.1.2 Monitoring Users

This is the phase of detecting terrorist users. Here, the textual content collected from a Web is modified into a vector called access vector, and it is compared with the 'typical terrorist behavior.' A predefined threshold is set, and whenever the similarity goes above the threshold, the alarm beeps. Contents other than textual contents (such as images) are considered absurd by the system.

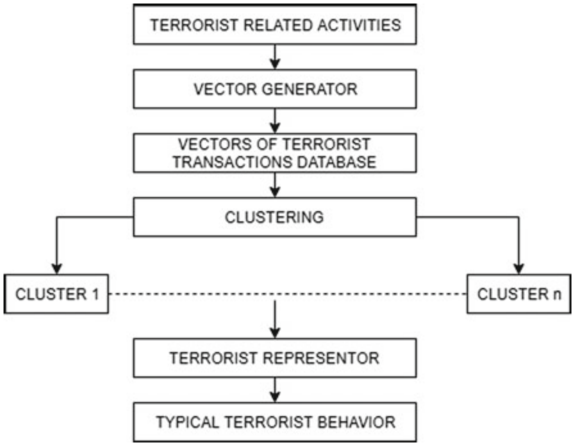
3.1.3 Learning Typical Terrorist Behavior

It represents the typical behavior of terrorist groups depending on the information collected about their activities on the Web. This is quite important since the information collected from the Web is used as input to the 'vector generator,' and there they are converted into vectors of weighted terms. Then, these are stored in vector of terrorist transaction database. This typical terrorist behavior helps us analyzing other profiles and to distinguish between a terrorist and non-terrorist profile (Fig. 1).

4 Detection of Typical Terrorist Behavior

The vector generator changes over the substance of each page obtained by a client into a vector portrayal (access vector) within the monitoring module. The detector utilizes the Typical terrorist behavior, and the obtained vector attempts to decide if the access vector has a connection with a terrorist. This is achieved by figuring closeness or similarity among the access vector and all centroid vectors of the Typical terrorist behavior. Similarity measurement can be done using the cosine measure. An alarm is raised when the similarity between the nearby centroid and the access vector crosses the predefined threshold (th). The similarity between the nearest centroid and access vector can be calculated as:

Fig. 1 Finding typical terrorist behavior flow



$$\text{Sim} = \frac{\sum_{i=1}^{p}(t\text{Ovi}1.t\text{Avi})}{\sqrt{\sum_{i=1}^{p} t\text{Ovi}1^2 \sum_{i=1}^{p} t\text{Avi}^2}}, \ldots, \frac{\sum_{i=1}^{p}(t\text{Ovin}.t\text{Avi})}{\sqrt{\sum_{i=1}^{p} t\text{Ovin}^2 \sum_{i=1}^{p} t\text{Avi}^2}}$$

where

Ovi	ith centroid vector
Av	Access vector
tOvi1	ith term in vector Ovi
tAvi	ith term in vector Av
p	Number of unique terms in each vector

When maximum of Sim increases the determined threshold value, i.e., max(Sim) > th, the detector raises an alarm.

This threshold is used to control detection sensitivity. Sensitivity of detection decreases with higher threshold value, thereby lowering the number of alarms, accuracy increases and false alarm number is reduced. Likewise, when the threshold value is low, sensitivity increases that increase number of alarms and accuracy is decreased. The user determines the optimal value of threshold according to personal preference.

5 Experimental Settings

A prototype system was built initially for evaluation of proposed detection methodology. The experiment included nine systems. Each computer is provided with a constant IP address and is assigned a proxy server, which represents an ISP and enables computers to access to Web. About 800 transactions were generated by 8 students who accessed Web sites concerned with general topics. Some other students were instructed to access terrorist-related Web sites, and around 214 transactions were generated. The server was already installed with vector generator, clustering, and detector module.

Vcluster program from the CLUTO Clustering Tool [10] is utilized to execute the clustering where the grouping calculation utilized was ‘k-way,’ and the similarity between articles was calculated utilizing the cosine measure. We use k means algorithm to find out the clusters from which the affected zones will be recognized.

5.1 Evaluation Measure

The suggested methodology should be better than existing methodologies and hence requires strong evaluation parameters. To evaluate the performance of the system, following measures can be used:

- True-positive rate (TP): the percentage of terrorist pages that crosses the predefined threshold (here, terrorist pages will be obtained from the users simulating terrorists.)
- False-positive rate (FP): the percentage of pages that are included in regular Internet access and determined incorrectly by the system as related to terrorist activities, i.e., the percentage of non-terrorist pages that crosses and the predefined threshold and suspected as terrorists falsely.

6 Result Summarization

Steps followed in the experiment are:

1. The value of the threshold is set to 0 initially, and number of clusters is 9.
2. 43 vectors are selected randomly out of 800 non-terrorist vectors which are used to test system performance.
3. Then, 43 vectors are again selected randomly out of 214 terrorist vectors which are used to test the detection ability of the system.
4. Remaining 80%, i.e., 171 out of 214 terrorist vectors, were used to train the system. A set of clusters related to terrorist topics were produced using clustering algorithm and centroid is calculated for each topic. Here, we get a set of k vectors that form our typical terrorist behavior.
5. Then, the true-positive rate is calculated, i.e., 43 vectors, related to terrorist users, are given as input to the detector, and the percentage of raised alarm is calculated.
6. False-positive rate is calculated by providing 43 vectors of non-terrorist users or regular users as input, and the percentage of mistakenly raised alarms is calculated.
7. Then, accuracy is calculated as number of alarms that were related to terrorist users divided by total number of alarms.

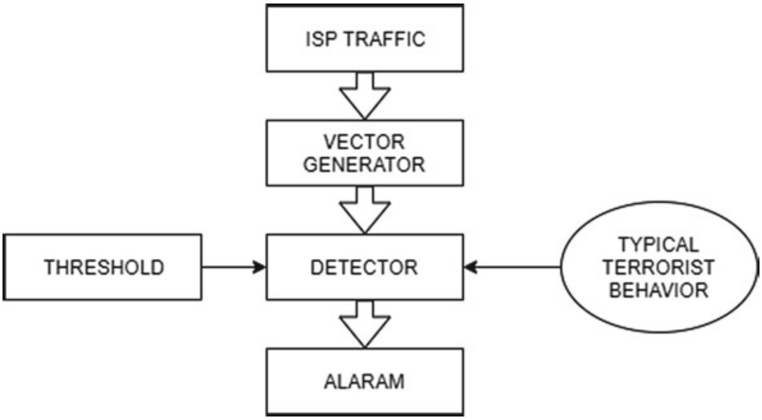


Fig. 2 Monitoring module

8. Steps 5–7 are repeated for different threshold values (between 0 and 1).
9. Steps 3–8 are repeated for five times, the average result of whose is shown in the ROC graph.
10. The entire process is repeated for different values of *k* to determine sensitivity of the system.

Here, we show the evaluation of the whole process using 5 and 9 clusters, i.e., $k = 5$ and $k = 9$. The *X*-axis in the ROC curve shows the false positive (FP), and the *Y*-axis represents true positive (TP). An average run of 5 runs is represented in each point of the ROC curve.

Figure 2 represents graph showing accuracy as a function of threshold which shows accuracy as monotonically increasing. However, the best suitable threshold value cannot be determined from the graph and hence depends completely on the user.

Our prototype obtained TP = 93% and FP = 11.7% on an average, whereas the ADMIT system command-level data of user produces TP = 70% and FP = 15% (Figs. 3 and 4; Table 1).

7 Deployment Issues

The law enforcement agencies can deploy the methodology in two different ways:

1. ISP-based system: Implementation of the proposed methodology is important here because it makes easy identification of suspicious user since IP is provided to user by ISP. But privacy issue is a major concern that comes into picture in this case. Also, ISP awareness is essential.

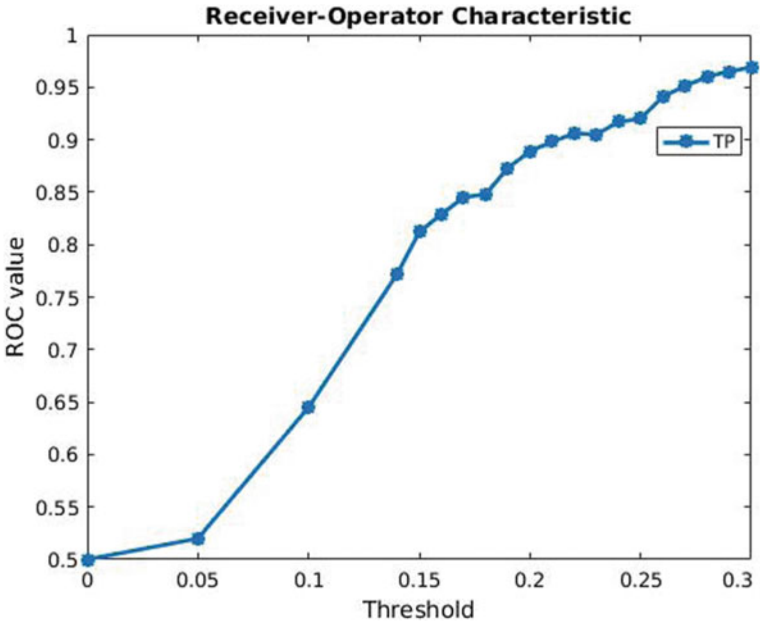


Fig. 3 Data accuracy

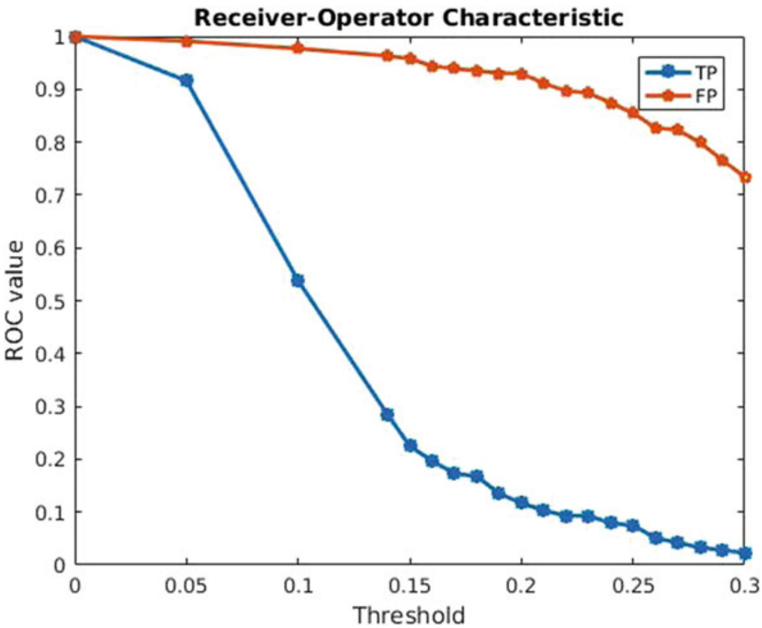


Fig. 4 ROC characteristic

Table 1 Result of TP, FP, and accuracy

Threshold	TP	FP	Accuracy
0	1	1	0.5
0.05	0.916	0.991	0.520
0.1	0.538	0.977	0.645
0.14	0.285	0.963	0.772
0.15	0.225	0.958	0.812
0.16	0.197	0.944	0.829
0.17	0.173	0.939	0.845
0.18	0.168	0.935	0.848
0.19	0.135	0.930	0.873
0.2	0.117	0.930	0.889
0.21	0.103	0.911	0.898
0.22	0.093	0.897	0.906
0.23	0.093	0.893	0.905
0.24	0.079	0.874	0.917
0.25	0.075	0.855	0.920
0.26	0.051	0.827	0.941
0.27	0.042	0.823	0.951
0.28	0.033	0.799	0.960
0.29	0.028	0.766	0.965
0.3	0.023	0.734	0.969

2. Network-based system: This new methodology raises the issue of privacy breach since it involves eavesdropping on communication lines that connect the Internet backbone with the ISPs.

8 Conclusion

Terrorism is an organized crime. Therefore, this field needs extensive research and major actions for controlling mass death and destruction. The paper presents an innovative methodology for detection of activities related to terrorism on the Web.

The research involves following issues

1. Representation of documents—This issue is important because favorable outcome of methodology depends upon the precision of content representation.
2. Detection methodology—Monitoring sequence of page views may give more accurate results than raising an alarm.
3. Optimal Settings—Determining some system settings such as predefined threshold for detection purpose and the number of clusters requires further more analysis.

The methodology provided here proves to be an effective system or method to check growing terrorism to some extent. This detection methodology can be applied in various other fields also, like detection of pedophiles whoever surfs the Web to access child pornography sites and other related criminals.

References

1. Gundabathula, V.T., Vaidhehi, V.: An efficient modelling of terrorist groups in India using machine learning algorithms. Indian J. Sci. Technol. **11**(15) (2018)
2. Goteng, G.L., Tao, X.: Cloud computing intelligent data-driven model: connecting the dots to combat global terrorism. In: 2016 IEEE International Congress on Big Data (BigData Congress), San Francisco, pp. 446–453 (2016)
3. Talreja, D., Nagaraj, J., Varsha, N.J., Mahesh, K.: Terrorism analytics: learning to predict the perpetrator. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, pp. 1723–1726 (2017). https://doi.org/10.1109/icacci.2017.8126092
4. Mahmood, T., Rohail, K.: Analyzing terrorist events in Pakistan to support counter-terrorism-Events, methods and targets. In: 2012 International Conference on 2012 Robotics and Artificial Intelligence (ICRAI), pp. 157–164
5. Corner, E., Gill, P., Mason, O.: Mental health disorders and the terrorist: a research note probing selection effects and disorder prevalence. Stud. Confl. Terrorism **39**(6), 560–568 (2016)
6. Gutfraind, A.: Understanding terrorist organizations with a dynamic model. In: Mathematical Methods in Counterterrorism 2009, pp. 107–125. Springer, Vienna
7. Chung, W., Tang, W.: Building a web collection for online surveillance of US domestic terrorism. In: 2012 IEEE International Conference on Intelligence and Security Informatics (ISI), pp. 195–195. IEEE (2012)
8. Perliger, A., Pedahzur, A.: Social network analysis in the study of terrorism and political violence. Polit. Sci. Polit. **44**(1), 45–50 (2011)
9. Crenshaw, M.: Current research on terrorism: the academic perspective. Stud. Confl. Terrorism **15**(1), 1–11 (1992)
10. Krebs, V.E.: Mapping networks of terrorist cells. Connections **24**(3), 43–52 (2002)
11. Horgan, J.: From profiles to pathways and roots to routes: perspectives from psychology on radicalization into terrorism. Ann. Am. Acad. Polit. Soc. Sci. **618**(1), 80–94 (2008)
12. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press (1994)
13. Mishra, N., Swagatika, S., Singh, D.: An intelligent framework for analysing terrorism actions using cloud. In: Patnaik, S., Ip, A., Tavana, M., Jain, V. (eds.) New Paradigm in Decision Science and Management. Advances in Intelligent Systems and Computing, vol. 1005. Springer, Singapore (2020)

Facial Expression Recognition System (FERS): A Survey

Sonali Mishra, Ratnalata Gupta, and Sambit Kumar Mishra

Abstract Human facial expressions and emotions are considered as the fastest way of the communication medium for expressing thoughts. The ability to identify the emotional states of people surrounding us is an essential component of natural communication. Facial expression and emotion detector can be used to know whether a person is sad, happy, angry, and so on. We can better understand the thoughts and ideas of a person. This paper briefly explores the idea of recognizing the computerized facial expression detection system. First, we have discussed an overview of the facial expression recognition system (FERS). Also, we have presented a glimpse of current technologies that are used for the detection of FERS. A comparative analysis of existing methodologies is also presented in this paper. It provides the basic information and general understanding of up-to-date state-of-the-art studies; also, experienced researchers can look productive directions for future work.

Keywords Facial expressions · CNN · Deep CNN · FERS

1 Introduction

Facial expression is one of the nonverbal communications for humans to convey their message. It plays a significant role in our daily life communication. Human facial expression and emotions play an important way for human being to socialize. They are the major channels of conveying social information among individuals. Facial

S. Mishra (✉) · S. K. Mishra
Siksha O Anusandhan (Deemed to be) University, Bhubaneswar, Odisha, India
e-mail: sonalimishra@soa.ac.in
URL: http://www.springer.com/lncs

S. K. Mishra
e-mail: sambitmishra@soa.ac.in

R. Gupta
Oriental University, Indore, India
e-mail: ratnalata@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_5

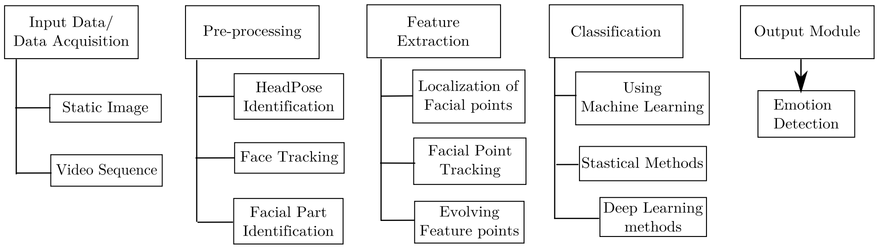


Fig. 1 Basic steps required for facial expression detection system

expression recognition is a complex and challenging problem. Therefore, automatic facial expression recognition system (FERS) is important nowadays and uses many applications such as monitoring, e-learning, medicine, entertainment, and law [1].

It is challenging to implement FERS in some areas like human–computer interaction as in health care [2], education [3], industry [4], entertainment, law, and crime. Numerous traditional FER systems require manual feature selection methodologies like histogram of oriented gradients (HoG), LBP, SIFT techniques, etc.) and then feed to a custom-designed classifier to classify expressions. However, such methods fail to produce accurate results because the attributes are extracted manually and also become difficult when the dataset is large. In general, FERS operates in three stages: preprocessing, feature extraction, and classification. Each stage of FERS has its own importance as well as complexity in the final decision step. The investigation of facial expression is the fundamental research area of psychology, and a large volume of research works has been published in recent times. However, the performance still requires potential improvement in order to meet the real-time scenario. Figure 1 describes the basic steps for the facial expression recognition system.

The rest of the paper is summarized in the following way. Section 2 briefly describes about the past researches carried out on facial expression recognition system (FERS). Section 3 introduces the details about available datasets used for FERS. A comparison of results of the past methods is given in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Related Work

In the past few years, different specialists have pulled into an automated computerized system [5] and added a different range of works until the present day in the area of FERS. Many researchers worked to overcome the problem of recognizing facial expressions. Also, research work has been conducted in the direction of creating an automated facial expression recognition system for the real-time environment. Pan et al. [6] have suggested a trainable deep neural network (DNN)-based frame that combines spatial and temporal information from the video. The proposed technique demonstrated an accuracy of 65.7 and 42.98% on RML and eNTERFACE05 dataset, respectively. The drawback of the suggested framework is that it has a huge parameter lists, and it takes higher computation time.

Minaee and Abdolrashidi [7] have proposed an end-to-end deep learning approach based on convolutional network. Experimental results obtained on FERG, FER-2013, JAFFE, and CK+ datasets denote an overall accuracy of 92.8%, 70.0%, 92.8%, and 98.0%, respectively.

Mokhayeri and Granger [8] have proposed a new technique to attain a high level of performance in video-based face recognition (FR). This reconstructs a probe image utilizing an auxiliary variational dictionary and an augmented gallery dictionary. Albrici et al. [9] presented a framework that exhibits automated facial expression recognition in videos. The pre-trained CNN model in a short video clip is dedicated to analyzing the video frames. Then, a LSTM network is designed to process the trajectories of the facial landmarks. The accuracy recorded for CK+ dataset was 97.4%, for MUG dataset 95.5%, and self-created dataset 80.2%.

Zhang et al. [10] have designed a hybrid deep learning model for FER in video sequences to learn the features effectively. Their proposed technique demonstrated an accuracy of 55.85, 73.73, and 71.43% on BAUM-1s, RML, and MMI datasets, respectively. The drawback of this model is that it has to be developed in a real-time FER. Liu et al. [11] have proposed an emotion recognition technique which combines audio–video and facial features in both non-temporal and temporal modes. RAF and FER2013 are the image datasets used to pre-train CNN models. Their proposed technique demonstrated an accuracy of 61.87%.

Xu et al. [12] have proposed an unsupervised auto-coder and Dempster–Shafer theory fusion method to combine different prediction results based on features. The entire dataset is divided into training set, validation set, and test set. The average accuracy of the model was 59.68.

Zeng et al. [13] have developed a combination of geometric and appearance characteristics of the face and are introduced to identify facial expressions due to possessing accurate and comprehensive information of emotions. For CohnKanade(CK+) database, the accuracy recorded was 95.79%. Sun et al. [14] have suggested a method to search three kinds of active regions, i.e., left eye regions, right eye regions, and mouth regions. CNN is used to extract information and classify expressions. Experiments are carried out on the CK+ database, JAFFE database, and NVIE database. The accuracy recorded was 95.36%, 96.57%, and 89.38%, respectively.

3 Dataset Used

There are sufficient data available on a public platform that incorporates many categories of possible situations for developing a facial expression recognition system (FERS). This section aims at describing the publicly accessible datasets used for

Table 1 An overview of image-based FERS datasets

Dataset	Samples	Subject	Expression distribution	Access
JAFFE [15]	213 images	10	6 basic and 1 natural expression	http://www.kasrl.org/jaffe.html
FER-2013 [16]	35,887 images	NA	6 basic and 1 natural expression	https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge
FERA [7]	188	10	5 expressions	https://ibug.doc.ic.ac.uk/resources/FERA15/
FERG [7]	55,767	6	6 basic and 1 natural expression	https://computervisiononline.com/dataset/1105138868
RaFD [17]	1608	67	6 basic and 1 natural expression	http://www.socsci.ru.nl:8180/RaFD2/RaFD
AffectNet [18]	450,000 images	NA	6 basic and 1 natural expression	http://mohammadmahoor.com/databases-codes/

Table 2 An overview of video-based FERS datasets

Dataset	Samples	Subject	Expression distribution	Access
CK+ [19]	593 image sequences	123	8 expressions	http://www.consortium.ri.cmu.edu/ckagree/
MMI [9]	740 images and 2900 videos	25	6 basic and 1 natural expression	https://mmifacedb.eu/
AFEW 7.0 [20]	1809 videos	NA	6 basic and 1 natural	https://sites.google.com/site/emotiwchallenge/
Oulu-CASIA [21]	2880 image sequences	80	6 basic expressions	http://www.cse.oulu.fi/CMV/Downloads/Oulu-CASIA
BAUM-1s [10]	1222 video clips	31	6 basic expressions and 4 mental states	https://archive.ics.uci.edu/ml/datasets/BAUM-1
eNTER FACE05 [6]	1290 videos	43	6 expressions	

designing FERS. Tables 1 and 2 present a summary of these datasets, consisting of the primary reference, number of cases, number of image or video samples along with the links to access those datasets.

4 Discussion and Comparison of Results

This section briefly discusses the recently developed methods in the field of FERS. The comparison of different techniques in various datasets is presented in Table 3. Experimental results showed in Table 3 indicate that the performance of convolutional neural network (CNN) obtained higher accuracy on CK+ dataset. We have also highlighted the recently proposed methods. Deep convolutional neural networks have been used for feature learning in face recognition. Figure 2 represents the accuracy comparison of different methods on CK+ dataset.

Table 3 Comparative analysis of existing methods

References	Classifiers	Dataset	Accuracy rate (%)
Pan et al. [6]	HMM, GMM, SVM	RML	65.72
		eNTERFACE05	42.98
Minaee and Abdolrashidi [7]	CNN	JAFFE	92.8
		CK+	98.0
Mokhayeri and Granger [8]	SRC	Chokepoint dataset	
		pAUC (20%)	0.8820.018
		AUPR	0.7450.019
		COX-S2V database	
		pAUC (20%)	0.895 0.020
		AUPR	0.766 0.017
Porcu et al. [22]	SVM	Webcam recordings	84:6
Albrici et al. [9]	Auto-encoder and SOM	MUG	95.5
		CK+	97.4
Zhang et al. [10]	Linear SVM	BAUM-1s	55.85
		RML+	73.73
Sun et al. [14]	CNN	CK+	95.36
		JAFFE	96.57
		NVIE	89.38

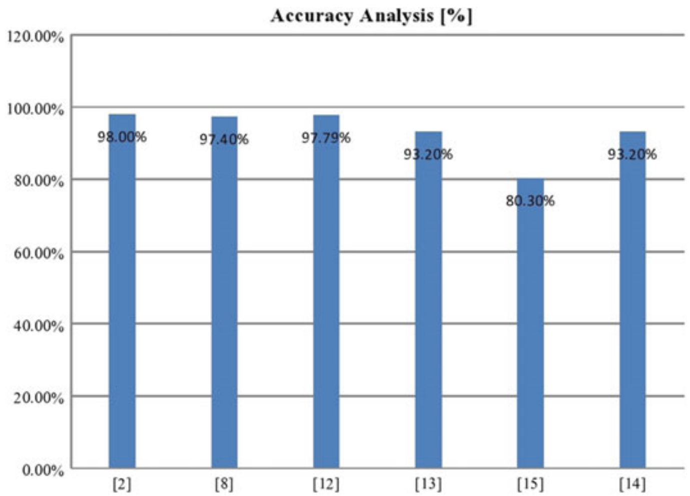


Fig. 2 Performance comparison of various methods on CK+ dataset

5 Conclusion

We have presented an overview of past developments in human facial expression and emotion detection. Several challenges can be taken care of in the field of an automatic emotion detection system. Some problems are such as (1) pose and frequent head movements, (2) the presence of structural components, (3) occlusion, (4) image orientation, (5) imaging conditions, (6) subtle facial deformation, and (7) ambiguity and uncertainty in face motion measurement. This concept must lead to a unique algorithm for dealing with the above problems. We have proposed model for recognition of FER and emotion recognition in video segments via a Horn–Schunck method for spatial and temporal features for fine-tuned CNNs, respectively. Further, newly developed methodologies can be used to design a more reliable technique for FERS in a better way.

References

1. Dong, J., Zheng, H., Lian, L.: Dynamic facial expression recognition based on convolutional neural networks with dense connections. In: 24th IEEE International Conference on Pattern Recognition (ICPR), pp. 3433–3438 (2018)
2. Girard, J.M., Cohn, J.F., Mahoor, M.H., Mavadati, S.M., Hammal, Z., Rosenwald, D.P.: Nonverbal social withdrawal in depression: evidence from manual and automatic analyses. Image Vis. Comput. **32**(10), 641–647 (2014)
3. McDaniel, B., D'Mello, S., King, B., Chipman, P., Tapp, K., Graesser, A.: Facial features for affective state detection in learning environments. In: Proceedings of the Annual Meeting of the Cognitive Science Society, vol. 29 (2007)

4. Assari, M.A., Rahmati, M.: Driver drowsiness detection using face expression recognition. In: IEEE International Conference on Signal and Image Processing Applications (ICSIPA), pp. 337–341 (2011)
5. Mishra, S., Majhi, B., Sa, P.K.: A survey on automated diagnosis on the detection of Leukemia: a hematological disorder. In: 3rd IEEE International Conference on Recent Advances in Information Technology (RAIT), pp. 460-466 (2016)
6. Pan, X., Ying, G., Chen, G., Li, H., Li, W.: A deep spatial and temporal aggregation framework for video-based facial expression recognition. IEEE Access **7**, 48807–48815 (2019)
7. Minaee, S., Abdolrashidi, A.: Deep-emotion: facial expression recognition using attentional convolutional network. arXiv:1902.01019 (2019)
8. Mokhayeri, F., Granger, E.: Robust video face recognition from a single still using a synthetic plus variational model. In: 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), pp. 1–8 (2019)
9. Albrici, T., Fasounaki, M., Salimi, S.B., Vray, G., Bozorgtabar, B., Ekenel, H.K., Thiran, J.P.: G2-VER: geometry guided model ensemble for video-based facial expression recognition. In: 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), pp. 1–6 (2019)
10. Zhang, S., Pan, X., Cui, Y., Zhao, X., Liu, L.: Learning affective video features for facial expression recognition via hybrid deep learning. IEEE Access **7**, 32297–32304 (2019)
11. Liu, C., Tang, T., Lv, K., Wang, M.: Multi-feature based emotion recognition for video clips. In: Proceedings of the 2018 on International Conference on Multimodal Interaction, pp. 630–634. ACM (2018)
12. Xu, Q., Sun, B., He, J., Rong, B., Yu, L., Rao, P.: Multimodal facial expression recognition based on Dempster-Shafer theory fusion strategy. In: 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia), pp. 1–5. IEEE (2018)
13. Zeng, N., Zhang, H., Song, B., Liu, W., Li, Y., Dobaie, A.M.: Facial expression recognition via learning deep sparse autoencoders. Neurocomputing **273**, 643–649 (2018)
14. Sun, A., Li, Y., Huang, Y.M., Li, Q., Lu, G.: Facial expression recognition using optimized active regions. Human-Centric Comput Inf Sci **8**(1), 33 (2018)
15. Lyons, M., Akamatsu, S., Kamachi, M., Gyoba, J.: Coding facial expressions with Gabor wavelets. In: Proceedings 3rd IEEE International Conference on Automatic Face and Gesture Recognition, pp. 200–205 (1998)
16. Goodfellow, I.J., Erhan, D., Carrier, P.L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.H., Zhou, Y.: Challenges in representation learning: A report on three machine learning contests. In: International Conference on Neural Information Processing, pp. 117–124. Springer, Berlin, Heidelberg (2013)
17. Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D.H., Hawk, S.T., Van Knippenberg, A.D.: Presentation and validation of the Radboud faces database. Cognit. Emotion **24**(8), 1377–1388 (2010)
18. Mollahosseini, A., Hasani, B., Mahoor, M.H.: Affectnet: a database for facial expression, valence, and arousal computing in the wild. IEEE Trans. Affect. Comput. **10**(1), 18–31 (2017)
19. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended Cohn-Kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pp. 94–101 (2010)
20. Dhall, A., Ramana Murthy, O.V., Goecke, R., Joshi, J., Gedeon, T.: Video and image based emotion recognition challenges in the wild: Emotiw 2015. In: Proceedings of the ACM on International Conference on Multimodal Interaction, pp. 423–426 (2015)
21. Zhao, G., Huang, X., Taini, M., Li, S.Z., PietikInen, M.: Facial expression recognition from near-infrared videos. Image Vis. Comput. **29**(9), 607–619 (2011)
22. Porcu, S., Uhrig, S., Voigt-Antons, J.N., Mller, S., Atzori, L.: Emotional impact of video quality: self-assessment and facial expression recognition. In: IEEE 11th International Conference on Quality of Multimedia Experience (QoMEX), pp. 1–6 (2019)

23. Song, X., Bao, H.: Facial expression recognition based on video. In: IEEE Applied Imagery Pattern Recognition Workshop (AIPR), pp. 1–5 (2016)
24. Shan, K., Guo, J., You, W., Lu, D., Bie, R.: Automatic facial expression recognition based on a deep convolutional-neural-network structure. In: IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA), pp. 123–128 (2017)

Load Balancing of Tasks in Cloud Computing Using Fault-Tolerant Honey Bee Foraging Approach

Bivasa Ranjan Parida, Amiya Kumar Rath, and Shrabanee Swagatika

Abstract Cloud computing is playing a vital role in the IT industries nowadays due to its enormous advantages of easy unlimited access to cloud services at any time, from any location with portability feature and reduced cost on hiring basis. The provision of various cloud computing services is facing challenges to implement efficient load balancing algorithms with fault tolerance facility in recent technology arena. In this paper, we proposed an algorithm of load balancing of tasks using honey bee approach with fault tolerance architecture. We have used CloudSim to simulate the algorithm and have done a comparative study with the existing one.

Keywords Cloud computing · Load balancing · Fault tolerance · Honey bee approach · Priorities of tasks

1 Introduction

The immergence of cloud computing has taken the world of IT into a new technical era.

This technology provides access to a collection of configurable computing resources over the Internet, like servers, storage, networks, applications and services on sharing mode, which reduces the time drastically to release resources with minimal management effort on pay-per-use strategy [1–3]. In the computing environment of cloud, clients need not to know the underlying design to utilize the services. Cloud

B. R. Parida (✉) · A. K. Rath
Department of Computer Science & Engineering, Veer Surendra Sai University of Technology, Burla, Odisha, India
e-mail: bivasranjanparida@gmail.com

A. K. Rath
e-mail: akrath_cse@vssut.ac.in

S. Swagatika
Department of Computer Science & Engineering, Siksha O Anusandhan University, Bhubaneswar, Odisha, India
e-mail: shrabaneeswagatika@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_6

is not just an accumulation of parallel and distributed systems like cluster and grid, but also it has various domains along with numerous virtualized nodes [4].

In this context, balancing the load among various nodes of the cloud is a tedious job for researchers, where nodes, neither will be overloaded nor under loaded in the network. A handful of studies regarding load balancing in cloud computing has been reported. Still efficient algorithms or models in load balancing are scarce. Even though the load balancing approaches in cloud environment have been becoming more and more sophisticated, they are not yet matured. The main lacunas are lack of skilled engineers to work in cloud architecture and lack of knowledge to create complex, efficient and qualitative algorithms, which can be updated quickly and reliably in dynamic cloud environment.

Furthermore, fault tolerance can be mentioned as the main restriction in these load balancing algorithms. In order to achieve robustness and reliability in cloud services, the failure should be detected and handled efficiently.

This paper has been sorted out as pursues. Section 2 describes the survey on the techniques to balance the load among the nodes in the cloud network, the categories of fault tolerance approaches in cloud computing along with our motivation of work to modify an existing load balancing algorithm. Our proposed work and the implementation are described in Sect. 3. Next, the conclusion and future work are mentioned in Sect. 4.

2 Literature Review

2.1 Load Balancing in Cloud Computing

The measure of work done by a computation system is called as load. It may be categorized on the basis of the capacity to store, CPU processing and network overhead. In the process of load balancing, overloaded nodes are found out, and afterwards, their excess load is migrated to under loaded nodes for improving the resource utilization and performance of the system [1, 5]. The major facts, which the researchers are considering, while designing an algorithm to balance load in the cloud environment, are, load estimation, load comparison, node selection, interaction between the nodes, system stability and performance, fault tolerance and nature of task to be transferred [1, 2]. The policies of load balancing algorithms are to determine when task migration should be done among the nodes, node selection for transferring the load based on its location and to acquire information about the nature of workload along with the average load on each node [2, 5].

The load balancing algorithms can be categorized based on various facts. On the basis of cloud environment, load balancing can be static, where load does not depend on the current state of the system using homogenous resources or dynamic, where current state of each node at run time is regularly supervised to adjust with the dynamic load requirements of the system having heterogeneous resources [1, 4, 6].

Dynamic load balancing can be done either in distributed or non-distributed fashion [5, 6]. Load balancing in distributed systems, where the algorithms are executed by all nodes, can be cooperative or non-cooperative based on the interaction between those nodes [5, 6]. The dynamic load balancing algorithms in non-distributed system are executed by single node or cluster of nodes to execute the process of load balancing [5, 6]. Again, these load balancing algorithms may be sender initiated, receiver initiated or symmetric [5]. On the basis of arrival of tasks onto the resources, the load balancing algorithms can be classified into immediate mode scheduling and batch mode scheduling.

While balancing load in cloud environment, we need to recognize and focus on some metrics to evaluate the performance of our algorithms. These metrics are throughput in terms of the number of tasks to be executed in a stipulated timeframe, scalability with any finite number of nodes, response time of the algorithm, fault tolerance ability, migration time of the tasks from one node to another, resource utilization, overhead due to task migration and inter process communication, energy consumption along with avoiding overheating, carbon emission, cost effectiveness and overall performance [1–3, 5–7]. To improve the performance of the system, throughput, scalability, fault tolerance and resource utilization should be improved or optimum, whereas response time, migration time, over head associated, energy consumption and carbon emission should be minimized [2].

2.2 Fault Tolerance in Cloud Computing

Fault tolerance is very much essential in cloud computing, to be reliable and robust in delivering the services efficiently, without wasting resource, energy and time. Fault tolerance can be proactive or reactive on the basis of procedures and policies. Prediction of fault or failure before execution is done in proactive fault tolerance, and the suspicious components are replaced in time, in order to avoid the occurrence of fault and recovery thereafter. But in reactive fault tolerance, the effect of fault that has already taken place is minimized or removed. Hence, in this policy, the remedial measures are taken after the occurrence of fault. The later one makes the system more robust. The quadratic minimum spanning tree can also be considered for an pair or path connect in minimum spanning tree for route selection for data balancing [8], but in case of a proactive system, this can be a costlier affair to choose between multiple paths after analysing a range of routes. A context and load aware methodology for efficient task scheduling using modified genetic algorithm known as family genetic algorithm [9] can also be used for a robust fault-tolerant system, but it may be time consuming for analysing the contest of the data.

Proactive fault tolerance techniques are self-healing, software rejuvenation and pre-emptive migration [10]. Divide and conquer method is implemented in self-healing fault tolerance mechanism, where the large task is divided into smaller sub tasks and run on different virtual machines as different instances of same application. If failure occurs, it automatically controls the failure. The system reboots or restarts periodically, in software rejuvenation, to make the system error free and clean. In pre-emptive migration, the applications in execution are supervised constantly by feedbackloop control mechanism, to avoid the occurrence of fault.

Different reactive fault tolerance mechanisms are check pointing, replication, job migration, S-guard, retry, task resubmission, timing check, safety bag check and rescue workflow, etc. [10]. Periodic check points are made to restart the execution from recent check point, instead of from the beginning, if failure occurs, resulting in saving time and resources with better throughput. In replication technique, multiple copies of the same task are generated and executed on various resources for successful execution with failure recovery. Due to any reason, if a specific machine fails during execution, then tasks of that machine are migrated to another working machine in job migration technique. S-guard is a roll back recovery mechanism. If any resource or task fails during execution, the resubmission of task is done on the same or any other cloud resource. When the system does not meet with the adequate safety properties, the blocking of commands takes place in safety bag check. A watch dog timer is used to check the in time execution of an application in timing check fault tolerance mechanism. Without rectifying the fault, rescue work flow technique allows the flow of application execution to persist until it will not be able to proceed.

2.3 Honey Bee Behaviour Inspired Load Balancing

The honey bee foraging behaviour from nature can be modelled and implemented in balancing the load in cloud environment. An algorithm named as honey bee behaviour-inspired load balancing (HBB-LB) is proposed by Dhinesh Babu et al. [11], in which, honey bee refers to the tasks removed from overloaded virtual machines (VMs) and the under loaded VMs are considered as food sources, to which these tasks are migrated for load balancing. Foraging of food source by a honey bee is synonymically considered for migrating and loading of a task to an under loaded VM. When the forager bees find abundant food source, they inform the other bees in the bee hive through their waggle dance. Similarly, the tasks removed from the overloaded VMs update the status of the VM with informations like the number of tasks being processed by the VM, and the details of high priority tasks currently executed by the VM. This algorithm focuses on balancing load efficiently across all the VMs with task priority taking into consideration, in order to minimize the waiting time of the tasks in the ready queue.

2.4 Motivation

Load balancing is one of the most important criteria in cloud computing. Effective assessment of the cloud nodes is one of the biggest challenges. Further balancing load with fault tolerance architecture is the need for real-world cloud applications, which cannot be ignored in recent scenario. Fault is not a regular activity. But if it takes place, then the cloud system losses its reliability and effectiveness. Hence, predicting fault before it takes place, and recovering from faulty situation is the biggest challenge for cloud researchers.

HBB-LB algorithm proposed by Dhinesh Babu L. D. et al. has the severe drawback of ignoring the occurrence of fault, which is a major concern in real-time cloud implementation. With this motivation, we concentrate upon load balancing in cloud environment with fault-tolerant approach.

3 Proposed Work and Implementation

CloudSim is the most efficient tool for modelling of cloud. Datacenters, hosts or servers, virtual machines and software are four components of a typical cloud. Load balancing method executes two major tasks, such as resource allocation and task scheduling. Mapping of the resources to the different components of cloud on demand basis is called as resource allocation, which is done in such a way that no node in the cloud should be overloaded. This mapping is done at two levels, such as VM mapping onto the server and task mapping onto the VMs. In contrast with resource allocation, task scheduling defines how the allocated resources will be available to the end users or will be executed on the processors, either fully or on sharing basis. This scheduling of tasks can be done either on space sharing or time sharing basis.

We have implemented Dhinesh Babu L. D. et al. proposed HBB-LB algorithm [11] for resource allocation on CloudSim and analysed different aspects thoroughly. We realized that the algorithm proposed by Dhinesh Babu L. D. et al. has the severe drawback of ignoring the occurrence of fault, which is a major concern in real-time cloud implementation. Hence, we incorporated the fault tolerance mechanism in our proposed algorithm, and therefore, we named our algorithm as fault-tolerant modified honey bee load balancing (FTMHBLB) algorithm.

We have called is Alive() upon task sets of various virtual machines on regular intervals to detect faulty VMs. We proceeded with reactive fault tolerance mechanism of job migration then after. Dhinesh Babu L. D. et al. have grouped VMs into three categories to trigger load balancing. The groups are over loaded VMs, under loaded VMs and balanced VMs. We have incorporated another group, faulty VMs as the fourth category to hold the VMs, which fail to operate and remain as dormant machines.

Dhinesh Babu L. D. et al. have removed the last task from the job queue of each VM and then on priority basis migrated the tasks to the appropriate under loaded VMs. Then, repeated the process till all the VMs are balanced. But we considered the high priority tasks only in the overloaded VMs and migrated them to those under loaded VMs, which contain minimum number of high priority tasks in their job queue for execution. Then, we calculated the standard deviation of load for each overloaded and under loaded VM to update the groups of over loaded, balanced and under loaded VMs. As all the high priority tasks are already migrated, the shortest

Table 1 Degree of imbalance before and after implementing FTMHBBLB algorithm

Number of tasks	Degree of imbalance before load balancing	Degree of imbalance after load balancing
15	0.2	0.07
20	0.12	0.03
25	0.11	0.02
30	0.1	0.02
35	0.09	0.01
40	0.076	0.01

tasks in the job queue of each VM are removed and migrated to the under loaded VMs on best-fit approach. This process is continued till the over loaded and under loaded VM groups become empty or we can say all the VMs are balanced.

We have designed our own algorithm FTMHBLB and implemented on CloudSim to test its effectiveness. We found it efficient in terms of balancing the load. Table 1 and Fig. 1 specify the load imbalance calculated before and after implementing FTMHBLB to justify the robustness of our proposed algorithm.

We have also compared the average waiting time of tasks keeping number of VMs constant for HBB-LB and our proposed FTMHBLB algorithm and found that our algorithm performs better. Figure 2 shows the comparison of average waiting time of tasks between HBB-LB and FTMHBLB.

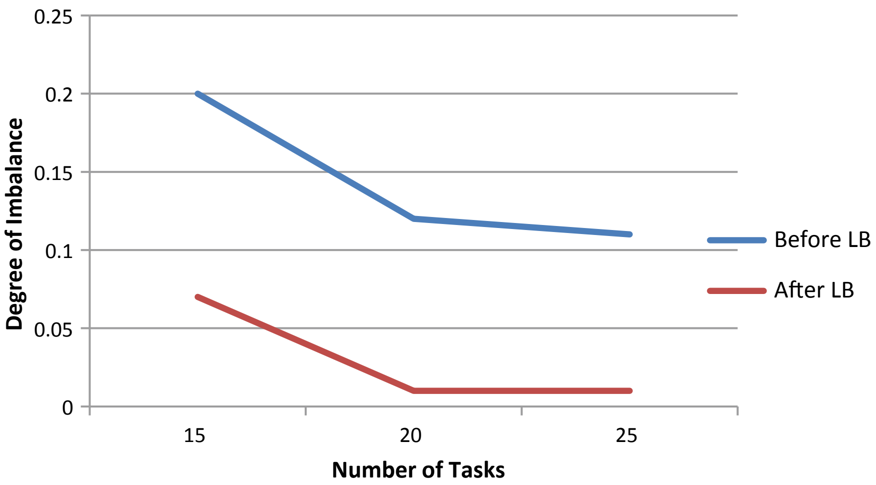


Fig. 1 Number of tasks versus degree of imbalance

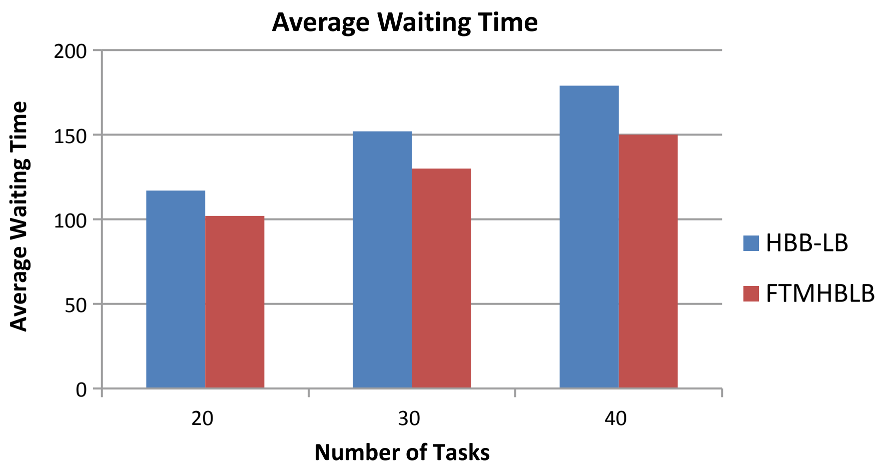


Fig. 2 Average waiting time comparison between HBB-LB and FTMHBLB

4 Conclusion and Future Work

Fault-tolerant modified honey bee behaviour-inspired load balancing (FTMHBBLB) algorithm outperforms in overall throughput as compared to HBB-LB with proper balancing of load and focuses on fault-tolerant mechanism. Thus, it reduces response time and makes the system reliable and robust.

In future course of action, we are interested to work on developing more fault-tolerant load balancing algorithm to achieve efficiency. Different kind of faults in real cloud environment can also be considered to make reliable algorithms to balance load among virtual machines effectively.

References

1. Desai, T., Prajapati, J.: A survey of various load balancing techniques and challenges in cloud computing. Int. J. Sci. Technol. Res. **2**(11), 158–161 (2013)
2. Dimri, S.C.: Various load balancing algorithms in cloud environment. Int. J. Emerg. Res. Manag. Technol. **4**(7), 263–266 (2015)
3. Dhami, M., Singh, J.: A review on presented load balancing techniques of cloud computing. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **5**(8), 647–650 (2015)
4. Katyal, M., Mishra, A.: A comparative study of load balancing algorithms in cloud computing environment. Int. J. Distrib. Cloud Comput. **1**(2), 5–14 (2014)
5. Kaur, R., Luthra, P.: Load balancing in cloud computing. In: Proceedings of International Conference on Recent Trends in Information, Telecommunication and Computing, pp. 374–381 (2014)
6. Micheal, A., Mehta, J.: Load balancing techniques in cloud computing. Int. J. Recent Innov. Trends Comput. Commun. **2**(10), 3038–3044 (2014)

7. Sidhu, A.K., Kinger, S.: Analysis of load balancing techniques in cloud computing. Int. J. Comput. Technol. **4**(2), 737–741 (2013)
8. Menaria, V.K., Jain, S.C., Nagaraju, A.: A fault tolerance based route optimisation and data aggregation using artificial intelligence to enhance performance in wireless sensor networks. Int. J. Wirel. Mobile Comput. **14**(2), 123–137 (2018)
9. Kaur, K., Kaur, N., Kaur, K.: A novel context and load-aware family genetic algorithm based task scheduling in cloud computing. In: Satapathy, S., Bhateja, V., Raju, K., Janakiramaiah, B. (eds.) Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing. Springer, Singapore 542 (2018)
10. Cheraghlou, M.N., Zadeh, A.K., Haghparast, M.: A survey of fault tolerance architecture in cloud computing. J. Netw. Comput. Appl. **61**, 81–92 (2016)
11. Dhinesh Babu, L.D., Venkata Krisna, P.: Honey bee behavior inspired load balancing of tasks in cloud computing environments. Appl. Soft Comput. **13**(5), 2292–2303 (2013)

Flip Learning: A Novel IoT-Based Learning Initiative

Sushil Kumar Mahapatra, Binod Kumar Pattanayak, and Bibudhendu Pati

Abstract Education is one of the basic things that encourage an individual to turn out to be among great people. Due to the emerging technology, the education system is also going to adopt these technologies. Hence, a lot of research work is going on to improve the educational teaching methodology using these cutting-edge technologies. Thus, to address this problem, a novel educational framework is proposed using Internet of things (IoT) as a key element. It represents a mixed learning approach which can improve conventional education framework with creative learning methodologies and advancements. Various gadgets and applications can be used to team up and share their thoughts among the partners (e.g., instructors and students) through the framework utilizing the services of IoT. Consequently, the proposed model can guarantee to get more brilliant and productive methodologies than the conventional ones.

Keywords Artificial intelligence · IoT · Education · Learning methodology

1 Introduction

Nowadays, advanced education in individual areas progresses because of the changing human advancement in adoption of technology in every field. There are various learning procedures and applications which are utilized to improve educational divisions. Conversely, restricted works have been done on this issue in developing countries. Internet of things (IoT) is an innovative technology that empowers

S. K. Mahapatra (✉) · B. K. Pattanayak
S 'O' A Deemed to be University, Bhubaneswar, Odisha, India
e-mail: m_sushil29@yahoo.co.in

B. K. Pattanayak
e-mail: binodpattanayak@soa.ac.in

B. Pati
Ram Devi Women's University, Bhubaneswar, Odisha, India
e-mail: patibibudhendu@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_7

inescapable collaboration in between individuals, items and the devices. It is a system-oriented methodology where information is gathered by installed sensors and sent to specific gadgets to facilitate activities. It is utilized in various areas, for example, human services frameworks, road safety management, academics, energy management system and so forth [1]. A collaborative and innovative learning technique called as Insight Learning of Things using IoT is presented to upgrade the nature of academics with its innovative technique. This platform demonstrates a synergistic learning and mixed learning methodology which is joined with a few interconnected modules. There are accumulated a tremendous measure of data from the partners' (e.g., instructors, students) advanced cells and wearable gadgets to produce important activities that can work together and share their thoughts, data utilizing IoT. Various kinds of programming and equipment-based modules like Web, mobile and implanted framework are utilized to improve the learning process in different nations. Different cutting-edge technology and services are incorporated here like cloud computing, student psychology, artificial technology, virtual and augmented reality. It is an adaptable as well as practical platform which oversees effectively various educational associations in the different countries.

2 Related Works

Many researchers have proposed on smart learning-based educational environment. A smart methodology has been proposed by the authors of [2] who incorporate the industry with the research institutes to enhance the teaching methodology for the master degree programs. In their proposed method, they have used the concept of flip classroom in the field of smart home and healthcare applications. They have also conducted a feedback survey which gives very impressive outcomes by using the above methodology. The authors of [3, 4] have identified some of the key features related to the smart classroom. They have also suggested some of the teaching and learning methodology based on the student feedback. A new methodology has proposed by the authors of [5] called Hstar in which both teacher and students can have an asynchronous communication with each other in real-time world specifically for software engineering course which incorporates with IoT, big data and cloud to solve the real-time problems. The smart gadgets used in IoT can fill the learning gap in the real-time environment to physical world [6]. This also enhances the exploratory learning environment for the students. Nowadays, the young generation is addicted toward the use of mobile phone for every purpose. Therefore, a new smart framework based on mobile apps for learning approach has been proposed called ALECSO for gulf countries [7]. According to the authors of [8–10], the learning outcomes can be enhanced and the cost of teaching can be reduced by using enhanced e-learning environment which is possible with the use of IoT in every aspect of the learning methodology.

3 Level-Based Approach

A level-based methodology is proposed where a few learning areas are considered for the upgrading the nature of advanced education in direct interaction mode with online-based learning system. Diverse specialized methods are engaged to upgrade the advanced education system progressively, which are depicted below:

3.1 Collaborative Learning Within the Campus

Presently, another education thought called smart classroom is executed where students get the substance as per their advantage. An ordinary classroom could be changed into a keen study room through the IoT innovations with conceptual analysis. The main objective of smart classroom is to advance customized improvement of learners, construct advanced characteristic and intelligent study hall outfitted within a get-together of a wide range of sorts of equipment and programming modules [11]. It underpins the understudies by giving all the study lecture notes and related materials including procedures of before, after and within the class. We propose a result-oriented focused learning technique of technological new era as opposed to conventional methodology. Community learning is a group-oriented idea that can assist the students in developing social collaboration aptitudes and invigorate basic reasoning. It incorporates instructing and learning that includes gatherings of students cooperating to take care of an issue. As indicated by mixed learning system, the proposed platform depends on savvy instructional method where flipped classroom (FC) is the significant concentration. Students can be benefited from both the online services and direct interaction for the educating and learning themselves through this platform. Instructor gives examination materials (e.g., lecture notes, e-notes and so on) through on line before flipped class. At that point, every class is partitioned into a few gatherings and each gathering needs to dole out a specific subject of fundamental class content and need to talk about this specific subject and introduce them in the class.

Learning analytics (LA) is a parametric-based analytic methodology utilizing some gadgets like camera, mouthpiece, sound sensor and so forth to observe student activity [9]. Through this technology, an educator can utilize LA to screen the learning procedure and distinguish issues of students. Conversely, student can likewise utilize LA to recognize their advancement and contrast their insights with different students.

3.2 Middle Product

In this level-based methodology, a few things are set to give reasonable services to stakeholders within and outside the campus. This area is known middle product that includes the following:

1. **Smart Contents**: Initially, instructor prepares the learning plan of the course. Generally, we use different study substances, for example, lecture notes, suggested and strengthening books, e-books and so forth. Furthermore, we recommend some smart books/presentations with enlarged real-time innovation so that the students can be skilled by seeing the tough points through 2D or 3D graphical overview through keen gadgets.
2. **Smart Devices and Apps**: Different contents are delivered to the students by smart IoT devices either from the cloud or utilizing a distributed arrangement. Learning materials can be gotten to locally on-grounds or internationally (off-grounds) utilizing savvy handheld lightweight versatile gadgets. The classroom materials can be shared and tendered utilizing IoT not only within the campus but also outside the study halls [6], and some of the mobile-based apps can be employed to learn, convey, collaboratively work together, question, reaction and offer data to both instructors and students using Internet.
3. **Smart LMF**: Smart learning management frameworks (SLMF) incorporate lectures handwritten materials and class contains, understudies assessments and appraisal, client input, gadgets and its utilization, online CMS, worldwide access approach and, smart objects and so forth. A LMF ought to be a brought together all framework where all other administration frameworks should be coordinated. Then again, students can likewise watch individual assessment result using their user ID. Moreover, students can likewise recover course contents from LMF. Plus, shrewd LMF requires monitoring a student's advancement in a few different ways, by the smart equipments like iris recognition or movement or outward appearance investigation such as facial expression and so forth.
4. **Smart Pedagogy**: A collected teaching methodology called smart pedagogy is to comprehend the student objectives, aptitudes, capacities and give ideal savvy learning condition to help students to accomplish their objectives. This instructional method depends on certain exercises including:

   (a) *Adaptive Teaching*: In this methodology, the instructor can adaptively change his class exercises according to the reactions and feedback of the students. At the point, when students are not inspired by investigation materials, they can pick their own arrangement.

   (b) *Self and Collaborative learning*: It is a group-based learning methodology which will assist the participants in developing social association aptitudes and animate the objective through analysis. A group of students will collaboratively do their projects and assignment in this learning methodology. A new self-learning technique is called content-based learning that permits the participants to watch video tutorials at their convenient place and talk about the content in study hall.

5. **Smart Assessments**: Some appraisal methodologies, for example, response of card indexing (CIR), various decision-making problems, i.e., multiple choice questions, can be utilized to legitimize student problem-solving capacity in a specific exercise. At the point, when an educator clarifies some theory in the study room, he wants the reaction of participants about perception of specific exercise [12]. Depending upon the understanding level, he sends some color to the online portal of educator. Nonetheless, there are additionally prescribed some different choices, for example, open finished, numerous decision questionnaires whose response can be selected by the Web-based applications as well as with the direct interaction with the instructor.
6. **Digital and e-Library**: Radio Frequency (RF) identifier tagged computerized repository is expected to get vital books as per the prerequisites. Here, the framework is equipped with PDA computing platform and RFID-tagged books stock management system synchronized with the central library database by the Internet. A smart mobile phone with CF card unit is employed for smart library inventory system which computerizes the registration, out procedures and tracks the information related to stock.

3.3 Customized Learning Out of the Campus

It is a criticism-based learning methodology where an instructing and academic condition is customized for individual students. In this way, specific viewpoints, for example, for every single student, course arranging and result, prospectus and course substance are reconstructed that relies upon the feedback given by the students. Each student has diverse arrangement of capacities and that has to be improved as indicated by his/her ranges of abilities. The following points are discussed about additional in the accompanying sections:

1. **Personalized Goal Planning**: The result needs to be tweaked as well as customized in customized learning condition. In this type of goal planning, the student's personal interest in a particular subject or course and his capability to take up that course has to be taken into consideration. The procedure of adaptable course plan starts with adaptability in goal planning to give customized learning.
2. **Customized Course Profiling and Syllabus**: All the students do not have a similar enthusiasm for each course, and it tends to be altered and arranged in a few sections and appraisal forms. As all the students have different competency levels and the course has to complete within the specified time, a set of customized problems can be given to the students to get the best result. Also, as indicated by certain researchers, the course educational programs ought to be structured and arranged 'in reverse' from the objective [5].

3. **Customized Digital Contents**: Generation of customized computerized course content is a significant part of customized learning system. Smart contents conveyance of these advanced materials can enable students to adapt quicker in the present technological world. Students can find out about contents through online study hall and coordinated effort applications whenever from anyplace.
4. **Customized Grading System**: A student will be assigned with problems, and the course materials will be given based on his progression and problem-solving capability by a customized grading system. The record of each student is followed consequently for assessment and future evaluation. Thus, in customized education, an adaptable reviewing framework is required which can be assisted by educational data mining and AI [13].
5. **Flexible Collaborative Works**: Bunch assignments are required to accomplish the most extreme effectiveness from a student. There are specific coordinated efforts required where students can meet online to conceptualize and take care of an issue together being on the Web. In this way, they need adaptability to pick colleagues and work in a gathering teaming up one another to tackle an issue. Furthermore, it tends to be done whenever anyplace off the grounds.
6. **Formative Assessment**: It is a continuous process of taking meaningful assessment that includes feedback from students after each session of the course, deciding along these lines the requirement of adaptive learning methodology. The regular monitoring of the student's general advancement in both on-campus and off-campus base is also essential [12].

4 Technical Aspects

In IoLT to enhance its level-based methodology of advanced learning system, some technological angles are considered which is adaptable. In this manner, we speak to related specialized perspectives quickly as pursues.

4.1 Mix of IoT and Cloud

IoT–cloud environment is required for present shrewd frameworks. As an educational foundation must have its very own server farm to safeguard every one of the information recorded from students, educators and different stakeholders consistently gather information from grounds utilizing IoT hubs. It is required to coordinate cloud and IoT guaranteeing elite and security [14]. The safe incorporation can be performed utilizing encryption or tokenization process. Cloud is laid over the server farm as a reflection and contains every one of the assets and study materials with the goal that it can convey utilizing CDN [15].

4.2 Information Management Framework

Information management incorporates recovering, examining, mining and storing of client information in a single framework. The effectiveness of information management depends for the most part on the use of the brought together framework. Customized learning structure improvement depends generally on the information management system too.

4.3 Content Delivery Network

It is a distributed server that delivers the user contents based on its geographic location. CDNs can oversee and appropriate enormous interactive media documents through IoT gadgets inside a specific system [15]. The administration additionally incorporates information sharing and steering, metadata gathering and load adjusting and so on, focusing on a progressively effective system. In any case, for same user content distribution, CDN can annihilate other blockage issue by using numerous servers. For guaranteeing high limit transmission capacity, effective content access managements, verification just as correction of information through replication, many proxy servers are employed here. To guarantee network availability all through the academic area, a very high-speed Wi-Fi network is a significant character. Wi-Fi routers or different remote passages ought to be accessible adequately as the IoT system and the content delivery network rely on it only.

4.4 Use of AI

A prediction model can be presented to enhance the educational outcomes through the utilization of educational data mining in order to analyze different aspects of the education system [13]. In the beginning, a set of questionnaires is prepared to collect information related to instructor and students' feedback, the course outcomes, the teaching methodology and students' attentiveness to the instructor's question. At that point, distinctive AI (ML) calculations are utilized to break down this information and concentrate critical components to upgrade this platform. This methodology is useful to take suitable choice to arrive at a well-characterized learning technique to the partners.

4.5 Virtual and Augmented Reality

A very innovative and joyful technique for the students is presenting the course material in the 3D model using virtual and augmented reality. Using this technique, the educational substance can be delivered through gaming. By utilizing this technique, the students can remember the content for a long time and their attentiveness about the course will grow up to a significant level.

4.6 Data Security in Network

Nowadays, students are very concern about the data security. Accordingly, a unified framework is required to give most extreme security and access policy toward the stakeholder's data [16]. In addition, for the customized learning, information with respect to students needs an efficient secure access policy and storage management technique. The identity of the stakeholder and access policy is very vital aspect that can be done through customized application in cloud [17]. As the student will carry their own smart gadgets to get into the network system, it is important to take legitimate estimations to guarantee system and data security.

5 Conclusion

This proposed work subtleties, the utilization of IoT to flourish an effective learning platform. The proposed platform is divided into three levels which are community-oriented learning, middleware and customized learning. Moreover, different accessible mechanical viewpoints are likewise examined to get an adaptable and financially savvy answer for developing nation. This examination demonstrates that IoT bigly affects higher educational part including lessening cost, efficient, advanced security and improved coordinated effort. It additionally speaks to an incredible potential impact to clients by structure virtual and customized relationship among them.

References

1. Bagheri, M., Haghighi Movahed, S.: The effect of the internet of things (iot) on education business model. In: 2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp. 435–441. IEEE Computer Society (2016)
2. Lena Claesson, J.Z.S.J., Lundberg, J., Hkansson, L.: Expert competence in remote diagnostics industrial interests, educational goals, flipped classroom & laboratory settings. In: 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), pp. 1–5. IEEE, (2018)

3. Vladimir, A.P.C.H., Uskov, L., Bakken, J.P., Rachakonda, R.: Smart pedagogy for smart universities. In: 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), pp. 1–5. IEEE (2018)
4. Shimaa Ouf, S., Ellatif, M.A., Helmy, Y.: A proposed paradigm for smart learning environment based on semantic web. Comput. Hum. Behav. **72**, 91–101 (2017)
5. Zhaoli Zhang, J.S.M.Z.H.L.Z.L., Cao, T.: Exploration of blended teaching pattern based on hstar and smart classroom (2017)
6. Federico Dominguez, X.O.: Smart objects in education: an early survey to assess opportunities and challenges (2017)
7. Mohamed Jemni, M.K.K.: The Alecso smart learning framework. In: Innovations in Smart Learning, pp. 91–101. Springer (2016)
8. Alexander Muriuki Njeru, M.S.O., Yi, S.: Iots for capturing and mastering massive data online learning courses. In: 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS) (2017)
9. Karri, S.A.V.U.C.H., Uskov, V.L., Bakken, J.P., Rachakonda, R.: Smart University: Conceptual Modeling and Systems Design, vol. 3(4). Springer International Publishing (2018)
10. Idris, M.Y., Stiawan, D., Habibullah, N.M., Fikri, A.H., Rahim, M.R.A., Dasuki, M.: Iot smart device for e-learning content sharing on hybrid cloud environment. In: 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), pp. 1–5. IEEE (2017)
11. Veeramanickam, M., Mohanapriya, M.: Iot enabled futurus smart campus with effective e-learning: i-campus. GSTF J. Eng. Technol. (JET) **3**(4) (2017)
12. Nicol, D.J., Macfarlane-Dick, D.: Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. Stud. High. Educ. **31**(2), 199–218 (2006)
13. Romero, C., Ventura, S.: Data mining in education. Wiley Interdisc. Rev.: Data Min. Knowl. Discov. **3**(1), 12–27 (2013)
14. Stergiou, C., Psannis, K.E., Kim, B.-G., Gupta, B.: Secure integration of iot and cloud computing. Future Gener. Comput. Syst. **78**, 964–975 (2018)
15. Vakali, A., Pallis, G.: Content delivery networks: status and trends. IEEE Internet Comput. **7**(6), 68–74 (2003)
16. Roy, S., Shovon, A.R., Whaiduzzaman, M.: Combined approach of tokenization and mining to secure and optimize big data in cloud storage. In: Humanitarian Technology Conference (R10-HTC), 2017 IEEE Region 10, pp. 83–88. IEEE (2017)
17. Shovon, A.R., Roy, S., Sharma, T., Whaiduzzaman, M.: A restful e-governance application framework for people identity verification in cloud. In: International Conference on Cloud Computing, pp. 281–294. Springer (2018)

Improvement of Load Balancing in Shared-Memory Multiprocessor Systems

Hasan Deeb, Archana Sarangi, and Shubhendu Kumar Sarangi

Abstract Parallel programming is one of the most effective approaches to handle complex problems regarding time complexity by reducing computation time, by getting the most of the capacity of the processors and shared-memory or distributed systems. One of the main ingredients of parallel programming is 'Loops' and especially DOALL loops. All loop scheduling algorithms try to achieve load balancing by using chunk resizing techniques (decreasing, increasing…). In this paper, prior loop scheduling algorithms will be evaluated on Mandelbrot. A new algorithm is obtained by merging the increasing and decreasing chunk size techniques in order to acquire the advantages of both approaches. The experimental results show that in the decreasing approach, a large number of small chunks in the last stages will increase the scheduling overhead because of the increasing of inter-processor communication. Also, the large chunk size of the initial stages will increase load imbalance. On the other hand, for the increasing approach, the large chunks assigned to processors at the last stages might increase load imbalance especially when iterations of the last stages are more time consuming than others. This work will introduce a new approach. This approach will try to minimize the load imbalance and communication overhead that are caused by using a decreasing chunk size approach and to minimize the load imbalance and scheduling overhead caused by the use of increase chunk size approach, which is going to provide better performance for many workload patterns.

Keywords Multiprocessor system · Shared memory · Distributed memory · Schedule methods · Load balance

H. Deeb · A. Sarangi (✉)
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India
e-mail: archanasarangi24@gmail.com

S. K. Sarangi
Department of Electronics and Instrumentation Engineering, ITER, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_8

1 Introduction

Parallel programming or parallel computing aims to execute a sequential program over multiple processors at the same time. Operations should be decomposed into sub-operations which can be executed on local or distributed processors simultaneously. The core idea of parallel compilers is to convert the sequential programs into parallel programs by transforming specific portions of programs into parallel portions without altering the program output. Here, many scheduling algorithms are used to guarantee the correct order of execution of parallel portions to guarantee the same result. One of the most common sources of parallelism in programs is loops. Operations executed in loops are repeated until loop conditions become false. Most of the scheduling algorithms focus on this part of a program when we want to parallelize it. As we mentioned, within the loop's body, the program executes sequence of operations multiple times. However, if the loop iterations do not depend on each other; we call it DOALL loop [1]. Independent iterations can be executed on multiple processors without any need to share data between them, which facilitate the management of scheduling in the parallel program. The best degree of parallelism can be achieved by distributing the DOALL loop's iterations among available processors equally which means each group of iterations needs the same time to be executed. The parallel version of the loop is called a parallel loop which is defined as a method that schedules the loop's iterations on multiprocessor systems as evenly as possible. Our study is concentrated on shared-memory multiprocessor systems where the scheduling strategy can be defined either by the programmer at compile time (static scheduling) or real time depending on selected algorithms (dynamic scheduling). In static scheduling, iterations are distributed among all processors and each processor has the same number of iterations. One disadvantage of this approach is load imbalance. On the other hand, dynamic scheduling is more suitable for load balancing, because it can deal with different time-consuming iterations and unknown loops' stop condition. Dynamic scheduling has two existing approaches: increasing chunk size approach and decreasing chunk size approach. Increasing chunk size approach attempts to achieve load balancing by assigning increased chunk size to subsequent scheduling steps. Also, decreasing chunk size approach tries to achieve load balancing by assigning decreasing chunk size, and the effect of communication overhead may not be a significant factor for a shared-memory machine, but it cannot be neglected in distributed memory platforms. There are three objectives of this research: The first goal of these objectives is to study various static and dynamic loop scheduling methods and see the effect of increasing number of processors on general performance of each algorithms. The second goal is to analyze the performance of increasing and decreasing of chunk size over Mandelbrot set. The third goal is to propose a new loop scheduling method that will use the increase and decrease techniques together. The idea behind that is to take the advantages of increase chunk size method in the initial stages and also take the advantages of decrease methods in the last stages. The new method starts with small size chunks, and then, there will be increasing in the size of chunks until we get a specific number of iterations; then, the size of chunks will be decreased until reaching the maximum number of iterations. In case of undefined number of iterations, we can specify a specific boundary number to switch the increasing size process to a decreasing process. There are four factors that should be taken into consideration in shared-memory multiprocessor systems, when we convert sequential program to parallel program: load imbalances, synchronization overhead, communication overhead, scheduling overhead. Different algorithms are suggested for dealing with these four factors, but it is difficult to handle them all at the same time. Each algorithm has its strength and its weakness. Therefore, this paper focuses on managing the distribution of parallel loop iterations among multiple processors in shared-memory multiprocessor systems evenly and minimizing overheads as much as possible.

2 Overview on Prior Algorithms and Techniques

2.1 Static Scheduling

In static scheduling, all iterations are distributed evenly among processors at compiling time. It is applied when the total number of iterations is known, the time required for each iteration is same and the compiler also knows the total number of processors available for use at compile time. Static scheduling can minimize the scheduling overhead but also cause load imbalance in case of iterations with different execution times. However, static scheduling performance may not be as expected when the chunks assigned to each processor are not equally distributed or if the loop's top condition is unknown at compile time. There are two types of static scheduling: **Block Scheduling**: Let N is the total number of iterations and P is the total number of processors. Each round consists of consecutive iterations and is assigned to one processor with size equal to $\frac{N}{P}$. In **Cyclic Scheduling**: In each round, one iteration is assigned to each processor. The next iterations are assigned in a cyclic fashion to the appropriate processor, i.e., iteration k is assigned to processor $k \bmod P$ which means execute k, $k + p$, $k + 2p$.

2.2 Dynamic Scheduling

In this approach, the scheduling is adjusted during the execution process. It is generally applied when the total numbers of iterations or processors are not known at compile time, or when the time required for executing each iteration is different.

Chunk Self-Scheduling (CSS): Each processor will pick up a chunk with some fixed size. Whenever a processor is idle; the number of iterations (k) is allocated to it, until all iterations are exhausted. When the chunk size is one, this scheme is considered pure self-scheduling (SS). If the size of chunk is equal to $\frac{N}{P}$, the scheme becomes static scheduling. This method has bad performance if the number of iterations is large and the chunk size is relatively small which will produce scheduling overhead.

Factoring: Guided self-scheduling (GSS) has one drawback which arises when large chunks are assigned to the first processors, and the remaining chunks need less time to execute. In this case, load imbalance occurs. Factoring solves this problem by allocating chunks in multiple stages. During each stage, the algorithm takes a subset of the remaining iterations and distributes them evenly between the processors. The subset iteration number is usually half of the remaining iterations. In each stage, the chunk size k allocated to each processor at step i is

$$k_i = \frac{R_i}{2 * p}$$

where $R_{i+1} = R_i - (p * k)$.

Decrease chunk size technique: As we mentioned before, dynamic scheduling methods move the scheduling decision from compile time to run time. CSS method tries to allocate the same amount of iterations to each processor. Another technique appears here; instead of allocating fixed chunk size to each processor at each step, chunks with different sizes are allocated. The first approach is decrease chunk size technique where the scheduling starts with large chunk size; then, the size of chunk reduces until we get chunk with specific number of iterations. Allocating a large number of iterations for the initial chunks will reduce the scheduling overhead. The small chunk size at the last scheduling steps will result in good load balance. In guided self-scheduling (GSS) method, the number of iterations assigned to each processor is decreased continuously at run time. At any point of execution, the size of the next chunk can be calculated with the formula as $C =$ (# remaining iterations)/(# processors). Allocating a large number of iterations subset to the initial chunks reduces the scheduling overhead and minimizes unequal finishing time. Allocating small chunk sizes provides good load balance at the end of scheduling steps.

Increase chunk size technique [2]: Here, we use increasing chunk size pattern to achieve load balance and reduce communication overhead at the final steps. By using decreasing chunk size pattern, extra scheduling overhead results. This overhead is caused by the small chunks we get at the last stages of scheduling. The major idea is to bundle smaller chunk into larger ones, so we can reduce the communication overhead and scheduling overhead. The main drawback of this approach is that it may lead to load imbalance if the last iterations are more time consuming than prior iterations. There are two suggested methods using this technique: fixed increase and variable increase. With fixed increase-scheduling (FI) case, we separate the scheduling process into multiple stages, where each stage has a number of scheduling steps. The number of stages is fixed, and the chunk size per stage will be increased until all iterations are exhausted. Once the number of stages has been fixed, the compiler or programmer needs to select the initial chunk size, and this selection is critical for optimal performance. Fixed increase has a good performance if all the iterations have same or fixed increment consuming time. But in variable increase-scheduling (VI) method, the compiler or programmer selects a constant value X such that an initial chunk size can be computed such that $C = N/X^*p$, where X is a constant value. Unlike fixed increase, where the fixed size increment is added per stage, VI adds a variable size increment per stage. After the initial chunk size has been selected, VI increases the chunk size similar to way factoring decreases the chunk size.

Increase Decrease chunk size technique: This technique uses the advantages of increase as well as decrease techniques. In the decrease chunk size approach, when we distribute the workload between processors, we have extra scheduling overhead in the last phases, where we have a lot of small chunks. In the other hand, when we assign too much work that needs more time to execute than the following iterations to the first processors, imbalance overhead arises. For the increase chunk size approach, when we increase the chunk size at the final steps of the scheduling process, we minimize schedule overhead and communication overhead. But some other drawbacks arise; the large chunks in the last stages might increase the load imbalance which will lead to clear delay. This might result in non-working processors while other processors are still working. Now, if the increase chunk size technique used to solve the extra scheduling overhead and communication overhead in the last steps and decrease chunk size technique used to solve the load imbalance, obviously, the performance will be increased and the drawbacks of each method will be minimized. The suggested method works as follows: In the initial steps of scheduling, a chunk with a specific size C_{initial} defined by the programmer at compile time will be allocated; then, at each step, the size of chunks increases as follows:

$$C_i = C_{i-1} + \left(C_{\text{initial}} \big/ 2 * p\right) \tag{1}$$

where i is the number of step. The predefined initial chunk size plays a very important role in the overall performance of the algorithm, so it should be selected carefully. The load imbalance which appears when we have more time-consuming iterations assigned to processers at the beginning of scheduling process will be reduced to its minimum value. The increasing process will continue until we reach a specific number of iterations usually half, and then, the increasing process is altered to decreasing process. In case of undefined bound of iteration number, we can specify a specific bound number to switch the increasing process to decreasing. We start decreasing the chunk size by the same way we have increased it in the increasing phase and with the same increment value. When we reach the last steps, the chunks assigned to the processors will be almost equal to the initial predefined chunk size that previously defined by the programmer. So, it must guarantee no schedule overhead or imbalance load. For best performance, the initial chunk size must be not very big or small to avoid scheduling overhead. However, the initial chunk size must be large enough when the number of total iterations is large and vice versa. If the time required to process each iteration is same over all the processers, this will guarantee to finish all of the assigned work at the same time, but when the time required to execute each iteration is different, then some processers may take more time to finish its work, while the other processors work on the next iterations. All the suggested methods try to solve this problem in different ways, and the prior increased chunk size techniques take the feature of ‘stages’ that proposed in factoring method, wherein for each stage, there is either fixed or increased chunk size. The new method works without stages, so the change in the chunk size will still occur until reaching the predefined alter or stop condition. In this case, if one of the processors stuck in a complicated operation, the other processors will execute the next works and they may finish together, but if the complicated iteration found at the last iterations, some load imbalance will appear.

3 Result and Discussion

To evaluate the performance of the new method, we compare it with the previously mentioned static and dynamic methods. The experiments have conducted over Mandelbrot set generating program and the results detailed accordingly.

3.1 The Mandelbrot Set [3]

Mandelbrot set is a set of points in the plane, defined as follows. Given a point (x, y), compute a sequence of other points (a_i, b_i), $i = 0, 1, 2, \ldots$ using the following formulas:

$$
\begin{aligned}
a_0 &= 0, b_0 = 0 \\
a_{i+1} &= a_i^2 - b_i^2 + x \\
b_{i+1} &= 2a_i b_i + y
\end{aligned}
$$

If each point in the infinite sequence (a_i, b_i) stays finite, then (x, y) is a member of the Mandelbrot set. If the sequence of points (a_i, b_i) shoots off to infinity, then (x, y) is not a member of the Mandelbrot set. A computer program for computing an image of the Mandelbrot set needs a different criterion for deciding whether a point is in the set; it is not possible to compute an infinite sequence of points and still get the answer in a finite time. It can be proven that if the point (a_i, b_i) ever exceeds a distance of 2 from the origin, then the sequence will inevitably shoot off to infinity. So, set a limit on the number of points, say 1000 points, and start computing the sequence. If (a_i, b_i) exceeds a distance of 2 from the origin before i reaches the limit on the number of points, then (x, y) is not a member of the Mandelbrot set (white point). If i reaches the limit before (a_i, b_i) exceeds a distance of 2, then (x, y) is assumed to be a member of the Mandelbrot set (black point) [4]. We should mention that the experiments occur on a laptop with corei3 CPU and 4 idle processors available for scheduling. The time required to generate Mandelbrot set by different methods is displayed in Table 1.

Table 1 Time required to generate Mandelbrot set using different scheduling methods

Schedule method		Th = 1	Th = 2	Th = 3	Th = 4	Th = 8
Sequential program	1st try	17,850				
	2nd try	17,820				
	3rd try	17,740				
	Average	17,803				
Static (block method)	1st try	9223	11,070	14,860	9130	17,820
	2nd try	9335	10,540	14,880	9100	17,700
	3rd try	9455	10,840	14,980	9010	17,780
	Average	9340	10,817	14,907	9080	17,767
Static (block scheduling)	1st try	7440	6297	7310	9189	17,662
	2nd try	7222	6217	7296	9234	17,670
	3rd try	7560	6300	7268	9268	17,580
	Average	7408	6247	7289	9230	17,639
Dynamic (chunk self-scheduling) $X = 10$	1st try	6288	6181	7390	9170	17,660
	2nd try	6111	6106	7350	9060	17,670
	3rd try	6145	6118	7320	9110	17,680
	Average	6181	6134	7353	9113	17,670
Dynamic (guided self-scheduling) $X = 10$	1st try	6102	6410	7610	9250	17,760
	2nd try	6135	6460	7470	9140	17,850
	3rd try	6155	6540	7450	9470	17,770
	Average	6130	6470	7510	9287	17,793
Dynamic (variable increase scheduling) $X = 10$	1st try	6300	6716	7553	9265	17,543
	2nd try	6311	6613	7336	9252	17,537
	3rd try	6394	6689	7674	9362	17,535
	Average	6335	6676	7521	9287	17,539
Dynamic (new suggested scheduling) $X = 10$	1st try	6137	6107	7197	9011	17,455
	2nd try	6132	6097	7175	9033	17,459
	3rd try	6114	6112	7159	8977	17,527
	Average	6128	6105	7177	9107	17,480

Depending on the prior experiment results, Fig. 1 represents the time required to generate Mandelbrot set using different scheduling methods over a different number of threads:

We can measure the performance improvement using speedup formula. Speedup is the ratio of execution time of a single processor to the execution time of the same program on multiple number of processor [5] (Table 2):

$$\text{Speedup} = \frac{T_s}{T_p}$$

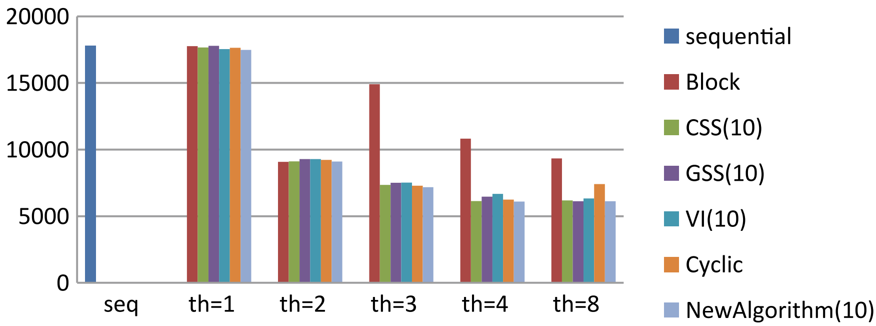


Fig. 1 Time required generating Mandelbrot set using different scheduling algorithms

Table 2 Asymptotic speedup of the program while increasing number of threads

Thread number	New method	Block	Cyclic	VI	GSS	CSS
$K = 1$	1	1	1	1	1	1
$K = 2$	1.95487	1.9606	1.92882	1.91698	1.91698	1.9535
$K = 3$	2.480563	1.19427	2.44245	2.36711	2.37057	2.4211
$K = 4$	2.905189	1.60142	2.849848	2.66671	2.75162	2.8558

Depending on the results, the speedup decreases when the number of threads exceeds 4. This happened because our test occurs on a device which has only four processors, and there is an extra scheduling overhead here to handle the increased number of threads (Fig. 2).

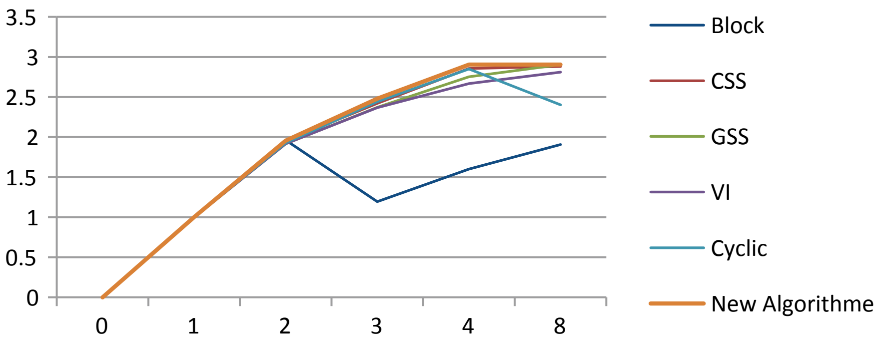


Fig. 2 Asymptotic speed up of Mandelbrot set on increased number of processors

4 Conclusion

The main goal of parallel programming is to optimally distribute workload between all available processors. In this paper, we examined the performance of previously proposed static and dynamic loop scheduling techniques. We also suggested a new method for DOALL loop scheduling. The new suggested method takes the advantages of increasing and decreasing chunk size techniques to get better performance in both shared and distributed memory systems; this will hopefully decrease both communication overhead and scheduling overhead and will achieve better load balancing. The new scheduling algorithm increases the size of chunks until it reaches a predefined bound. The algorithm will then alternate the increment by decreasing to return back to the predefined initial chunk size. In case of an unknown number of iterations, the alternating bound can be defined at compile time. Mandelbrot set was selected to experience and compare the existing algorithms with the suggested one. According to the experimental results, the new algorithm shows better performance in load balancing as well as reduction of communication and scheduling overhead. Although the results are not as good as expected compared with the prior algorithms, but we cannot judge its performance before it gets tested on distributed systems, where the communication overhead plays an important role in general system performance. As a future goal of this research, we will try to test our algorithm on distributed systems and we will try to find better ways to handle unknown number of iterations in order to reduce the role of programmer at compile time and avoid predefined constants (like the initial chunk size), although the step size of increasing and decreasing phases needs more experiments and testing to find the optimal value of it.

References

1. Fann, Y.W., Yang, C.T., Tseng, S.S., Tsai, C.J.: An intelligent parallel loop scheduling for parallelizing compilers. J. Inf. Sci. Eng. **16**(2), 169–200 (2000)
2. Philip, T., Das, C.R.: Evaluation of Loop Scheduling Algorithms on Distributed Memory Systems. Pennsylvania State University, Department of Computer Science and Engineering, College of Engineering (1996)
3. Devaney, R.L.: Unveiling the Mandelbrot set, Plus. Maths. Org. x, 1–17 (2006)
4. Kaminsky, A.: Building Parallel Programs: SMPs, Clusters & Java. Cengage Learning (2009)
5. Hwang, K., Jotwani, N.: Advanced Computer Architecture, 3rd. McGraw-Hill Education (2016)

Building Amazon Web Services Infrastructure from the Scratch for an Organization

Muskan Garg, Harsh, Harshit Singh, Shrabanee Swagatika, and Smita Rath

Abstract The Amazon Web services (AWS) infrastructure is very much essential for a newcomer for an organization for its usability and requirement for work environment. In order to have the adequate knowledge to build the AWS infrastructure here, in this project, we mainly focus on such kind of employees to help them to create the AWS infrastructure with ease. We also discussed various aspects and challenges while designing the AWS infrastructure. Here, in this paper, we focus on some of the basic steps to create as well as guide to creating the AWS infrastructure. We also provide the code as well as a flowchart model for a better explanation of the building work.

Keywords Virtual private cloud (VPC) · EC2 instance · Amazon web services

1 Introduction

AWS delivers scalable cloud computing with high availability and dependability, providing the tools that enable customers to sum a wide range and application. It

M. Garg (✉) · S. Swagatika · S. Rath
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: muskangarg273@gmail.com

S. Swagatika
e-mail: shrabaneeswagatika@soa.ac.in

S. Rath
e-mail: smitarath@soa.ac.in

Harsh · H. Singh
Department of Computer Science and Engineering, Gandhi Institute for Education and Technology, Bhubaneswar, Odisha, India
e-mail: harsh88588@gmail.com

H. Singh
e-mail: harshitsingh7873@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_9

helps the customers system to protect the confidentiality integrity and many more and the data is of the most important to Amazon Web services, as it maintains customer trust.

Reason to choose Amazon Web services as your cloud platform:

- Location: AWS would be the worldwide leader in cloud computing. It has 44 availability zones within 16 geographical regions around the world.
- Multi-region backups: AWS have the backup methods which will not affect their files from any natural or manmade disaster.
- Autoscaling: Customer can manage the application instances up and down for the unexpected demand. Autoscaling leads drives more efficiency.
- Payment and costs: As AWS provide autoscaling and user can start or stop instances, when needed. Customer can access necessary storage and many more services; so they need to only pay what services they use.
- Security: AWS provide full security to secure your infrastructure. It provides security in lower cost. It also takes care about your privacy as stored in AWS data center. Here, size of the data, files does not matter, it provides highest standard of world-class security.

About VPC—for the virtual servers—VPC provides an isolated virtual network.

AWS is a secure cloud services platform provide database storage, computer power, content delivery, analytics application, with lower IT costs and scale applications.

Villamizar et al. proposed a Web service consisting of monolithic and microservice architecture for cloud customer and provider to counter the various issues in cloud-like infrastructure and development cost [1]. Khmelevsky in his paper [2] provides a summary on efficient resource utilization in cloud for handling academic activities. Serrano provides a brief description on current status in cloud computing [3]. Varia describes the issues in traditional applications and shows the advantages of using cloud applications in the ever-growing data sets [4]. Bermudez et al. use Amazon Web services that provide storage, computing and content delivery services [5].

2 Proposed Work and Implementation

Working Model with Implementation explanation

Steps for building AWS infrastructure using scratch are shown in Fig. 1.

Step 1

Go to 'VPC' service, click on 'create VPC', assign Internet Protocol (IP) range, i.e., (10.0.0.0/16), give any name to the VPC then click on 'create' option.

Step 2

Here, we are creating three Subnets in which two will be public Subnets and one will be private Subnet. For first Subnet, i.e., Subnet 1, Go to 'Subnet' option, click on 'create Subnet', select VPC name (which we had given in our first step). Assign

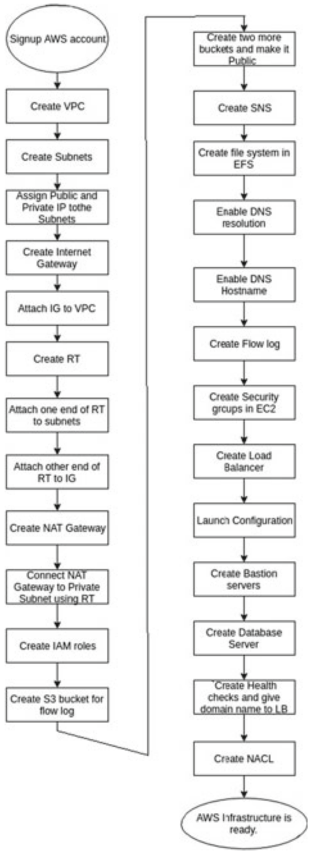


Fig. 1 Flowchart of the working model

IP address to the first Subnet, i.e., 10.0.1.0/24, then give name to the Subnets and then click on 'create' option.
For second Subnet, i.e., Subnet 2 follows same steps which are used to create first Subnet.
But in place of IP range, assign IP range to 10.0.2.0/24.
For third Subnet, i.e., Subnet 3 follows same steps which are used to create first and second Subnets. But in place of IP address, assign range to 10.0.3.0/24.1

Step 3

Here, we are assigning public IP to Subnet 1 and Subnet 2 and private IP to Subnet 3 as we will install database in Subnet 3. To assign public IP, Go to Subnets then select the Subnets one by one which you want to assign public IP then click on 'action' option then click on 'modify auto-assign IP setting' then select 'auto-assign IPV4' then click on 'save' option.

Step 4

Go to 'Internet Gateways (IG)' then click on 'create IG' option, give a name to IG and then click on 'create' option.

Step 5
Go to 'IG', select IG which you want to assign to VPC, click on 'action' option then select 'attach to VPC' option then select VPC and click on 'attach' option.
Step 6
Go to '(Route Table) RT' then click on 'create RT' option, give any name to RT then select VPC and click on 'create' option.
Step 7
Select 'created RT' and then go to Subnet associations, click on 'edit Subnet associations' then select the Subnets which we had assigned public IP then click on 'save' option.
Step 8
Go to 'RT' and select 'RT' then click on 'Route' option, click on 'edit Route' option then in the 'destination' option assign IP, i.e., 0.0.0.0/0 and inside the 'Target' option select IG, inside IG, select 'created IG' then click on 'save routes'.
Step 9
Go to 'Network Address Translation (NAT) gateway', click on 'create NAT gateway', and select Subnet, i.e., first public Subnet having IP 10.0.1.0/24. Then, create new Elastic IP by clicking on 'create New Elastic IP' option then Click on 'create NAT gateway'.
Step 10
To provide Internet to private Subnet, we have to connect NAT gateway to private Subnet. For this, we are using existing 'RT', i.e., default 'RT' of created 'VPC' by changing its name to 'NATRT'. For this, go to 'RT' and change the name. Then, connect one end of RT to NAT and another end to private Subnet. To connect one end of RT to NAT, select existing RT then click on 'Route' option, then click on 'edit route' then click on 'add route' then select created 'NAT gateway' with IP (0.0.0.0/0), then click on 'save route' option. Connect another end to private Subnet. Go to 'RT' and select it then select 'Subnet Association' option then click on 'edit' option and then select 'private Subnet' option and then click on 'save' option.
Step 11
Go to 'Roles' option then click on 'create roles' then select EC2 from the List, then click on 'next permission' then search the policy 'Simple Storage Service (S3) full Access' in the 'policy name' then click on 'Next Tags' and give the name to the role and click on 'Create roles.
Step 12
Go to 'S3' service then click on 'Create bucket' option and give any name to the bucket then click on 'next' option. Then, go to 'set permissions' then deselect 'block all public accesses to make the bucket public and click on 'next' option then click on 'create bucket' option.
Step 13
Go to 'S3' service and follow the steps of 'Step 12'.
Step 14
Go to 'Services' and click on 'Simple Notification Service (SNS)' then click on 'create Topic' any give a name to the topic then click on 'create' option.

Now creating ‘subscription’. For this, click on ‘create subscription’ then inside the ‘protocol’ option select the medium through which we can get the ‘notifications’ and we are using the ‘mail id’ to get the notifications, then click on ‘create subscription’ option and confirm the subscription by clicking the confirmation link sent to the given mail id.

Step 15

Go to ‘Elastic File System (EFS)’ service then click on ‘Create File System’ option and select the created ‘VPC’ then click on ‘next step’ and give the name then click on ‘next step’ and click on ‘Create File System’ option.

Step 16

Go to ‘VPC’ service and select the created ‘VPC’ then click on ‘action’ option then select the ‘Edit Domain Name System (DNS) resolution’ and enable it, then click on ‘save’ option.

Step 17

Go to ‘VPC’ service and select the created ‘VPC’ then click on ‘action’ option then select the ‘Edit DNS Hostname’ and enable it, then click on ‘save’ option.

Step 18

Go to ‘VPC’ service and select the created ‘VPC’ then click on ‘action’ option then click on ‘Create Flow Log’ option then inside the ‘filter’ option select all the options given there, then in the destination select ‘send to an S3 bucket’ option and inside ‘S3 bucket Amazon Resource Name (ARN)’, Enter the created S3 bucket ARN for the flow log. To get ‘bucket ARN’, go to ‘S3’ service, select the bucket and then click on

‘Copy ARN’ option and paste it on a box shown in the screen.

Step 19

Go to ‘EC2’ service then select ‘Security Groups (SG)’ option then click on ‘Create SG’ option and assign the name. In ‘Inbound’ option click on ‘Add rule’ and select

‘Secure Shell (SSH)’ and ‘http’ in ‘source’ option, then select ‘anywhere’ option and no operations on ‘Outbound’ option would be done then click on ‘create’.

Step 20

Go to ‘EC2’ service and select ‘Load Balancer’ option then click on ‘Create Load Balancer (LB)’ option, then select the LB type as per the requirement then click on ‘create’ option and give a name to LB. Select ‘VPC’ and both public Subnets in availability zones then click on the ‘next’ option and again click on the ‘next’ option then select both default and created SG and then click on ‘next’ option. Inside ‘configure routing’ option, give the target name and inside ‘Health check’ option, fill every option as per the requirement, then click on ‘register target’ and click on ‘next’ option and then click on ‘create’ option.

Step 21

Go to ‘EC2’ service, select ‘Launch Configuration’ option then click on ‘create launch configuration’ option then select the ‘Amazon Linux’ or any other as per the requirement, give a name ‘LC’, then inside ‘Identify and access Management (IAM) role’ select the created ‘role’ then click on ‘advanced details’ and inside the ‘user data’ option, write the command, i.e., called ‘BASH SCRIPT’. Now select storage as per the requirement then inside the ‘SG’ option click on ‘select an existing Security group’ option and select both default as well as created ‘SG’, then click on ‘create’ option and then click on ‘create an autoscaling group using the launch configuration’ option. Then, give a name to the ‘group’ and inside the ‘group size’ option enter the required no. of instances as these instances will work as a Web server, then inside the ‘Subnet’ option, select both ‘public Subnets’ then click on ‘next’ option and then click on ‘Add notification’, select the created ‘SNS’ then click on ‘next’ option then click on ‘review’ and then click on ‘create’ option.

Step 22

Go to ‘EC2’ service and click on ‘launch instances’ option then select Amazon Linux Amazon Machine Image (AMI) and also select the required instance. Inside the ‘configure instance details’ option select the created ‘VPC’ and first public Subnet having IP 10.0.1.0/24, then click on ‘next’ option then select the required storage then click on ‘next’ option then inside the ‘SG’ option select SSH and in source select ‘myIP’ then click on ‘review and launch’ option and then click on ‘launch’ option, then download ‘pem’ file or use existing pem file to open the instance.

Step 23

Go to ‘EC2’ service and click on ‘launch instance’ then select ‘Amazon Linux AMI’ and select required instance type. Inside the ‘configure instance details’ option select the created ‘VPC’ and private Subnet having IP 10.0.3.0/24, then click on ‘next’ option then select required storage then click on ‘next’ option then inside the ‘SG’ option select SSH, for SSH enter private IP of the ‘Bastion’ server then click on ‘Add rules’ and select ‘mysql’ and write IP address of first public Subnet, i.e., 10.0.1.0/24 in ‘IP Address’ option and again select ‘mysql’ and write IP address of second public Subnet, i.e., 10.0.2.0/24 then click on ‘review’ option and then click on ‘launch’ option.

Step 24

Go to ‘R-53’ and click on ‘DNS management’ option then click on ‘Health check’ option then click on ‘create health check’ option and assign a required name to the health check. Assign ‘Domain name’ to the LB, then inside the ‘path’ option give path, i.e., ‘index.html’. Then click on ‘advanced’ option and select ‘fast (10 s)’ option and fill up the other required options, then click on ‘next’ option then click on ‘yes’. Inside the ‘create alarm’ option select created ‘SNS’ topic and click on ‘create health check’.

Step 25

Go to ‘VPC’ and click on ‘Network ACLs’ option, click on ‘create network ACL’, give name, inside the ‘VPC’ option select created ‘VPC’ and then click on ‘create’ option. Now attach Subnets to NACL. For this, select created ‘NACL’ click on ‘Subnet association’ option, click on ‘edit Subnet association’ option, select all public Subnets then click on ‘edit’ option. After that click on ‘Inbound rules’, click on ‘edit inbound rules’ option, click on ‘add rules’ then inside the ‘Rule #’ option, enter the rule number. We are going to create 3 rules. For first rule, select ‘SSH’ inside the ‘type’ option and enter the IP inside the ‘source’ option.

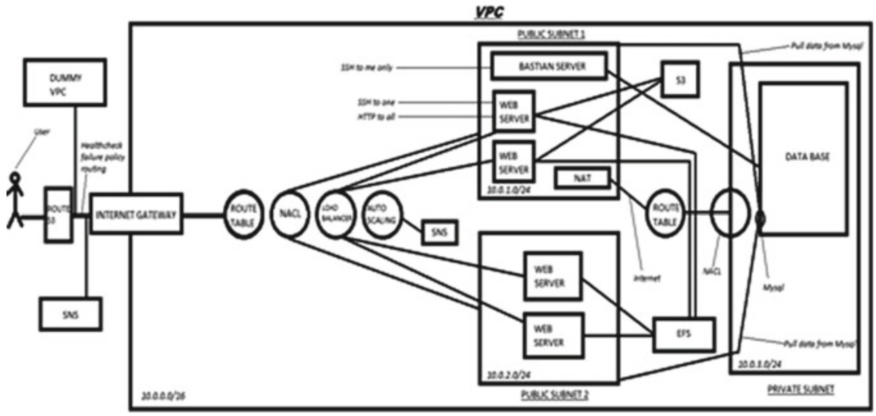


Fig. 2 AWS infrastructure of the model

For second rule, select 'HTTP' inside 'type' option and enter 0.0.0.0/0 inside 'source' option and third option is for 'Ephemeral ports' so select custom 'TCP rule' inside the 'type' option and enter 1024-65,535 inside the 'port range' option and enter 0.0.0.0/0 inside the 'source' option. Then, we need to allow same rules for 'Outbound rules'. For this, click on 'outbound rules' click on 'edit outbound rules' then add some rules which is used in 'inbound rules'.

Step 26

Our AWS infrastructure is ready as shown in Fig. 2.

Figure 2 depicts the overall infrastructure obtained from the working model.

3 Conclusion and Future Work

In this project, we have created AWS infrastructure from the scratch. Here, we included VPC, EC2, S3, IAM, SNS, EFS, LB, Route 53(R-53), NACL, Autoscaling, Dynamo Database (DB), Flow Logs, NAT, Red shift, Health check, RT, Bastion. From these services, we can build the infrastructure. There are so many temporary challenges that may arise at the time of this infrastructure creation like downtime, backup protection and limited control which can be dissolved within some time and they can be taken as our future work.

References

1. Villamizar, M., et al.: Infrastructure cost comparison of running web applications in the cloud using AWS lambda and monolithic and microservice architectures. In: 2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). IEEE (2016)
2. Khmelevsky, Y., Voytenko, V.: Cloud computing infrastructure prototype for university education and research. In: Proceedings of the 15th Western Canadian Conference on Computing Education. ACM (2010)
3. Serrano, N., Gallardo, G., Hernantes, J.: Infrastructure as a service and cloud technologies. IEEE Softw. **32**(2), 30–36 (2015)
4. Varia, J.: Best practices in architecting cloud applications in the AWS cloud. In: Cloud Computing: Principles and Paradigms, vol. 18, pp. 459–490. Wiley Press, New Jersey (2011)
5. Bermudez, I., et al.: Exploring the cloud from passive measurements: the Amazon AWS case. In: 2013 Proceedings IEEE INFOCOM. IEEE (2013)

Fault Detection for VANET Using Vehicular Cloud

Biswa Ranjan Senapati, Santoshinee Mohapatra, and Pabitra Mohan Khilar

Abstract Vehicular Ad hoc NETwork (VANET) is the meteoric growing research area in the field of academics and industry due to the technological advancement in sensing, computation, communication, and radio wireless technology. VANET is becoming so popular that it is widely used for various applications like safety application, convenience application, commercial application, productive application, and vehicular cloud service application. All these applications for VANET through routing will not be successful if the communication unit of the vehicle, i.e., On Board Unit (OBU) will be faulty. This paper proposed an automatic fault detection mechanism for the communication unit of the vehicle through vehicular clouds. The faulty detection of OBU is done at the base station. Cloud service is used for the transmission of data from the vehicle to the base station. The parameters like Fault Detection Accuracy (FDA) and False Alarm Rate (FAR) are used for the validation of this routing protocol.

Keywords VANET · Vehicular cloud · FDA · FAR · OBU

1 Introduction

Instruments or components like electronic components, network components, mobile components, etc. are subjected to failure because of fault due to excessive usage. The presence of a faulty component reduces the overall performance of the devices or the network. The communication unit in a vehicle in VANET is not the exception.

VANET is a subclass of Mobile Ad hoc NETwork (MANET) which consists of two components which are Road Side Unit (RSU) and vehicles [1]. For luxurious life,

B. R. Senapati (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: biswa.rnjn@gmail.com

S. Mohapatra · P. M. Khilar
Department of Computer Science and Engineering, National Institute of Technology Rourkela, Rourkela, Odisha, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_10

reduction in transportation time, and due to technological advancement the demand for the number of vehicles and manufacturing of vehicles is increasing day by day.

Figure 1 shows the number of vehicles manufactured per year [2]. Also, due to technological development in the electro-mechanical-home science department helps in the development of various sensors at an affordable cost for different applications. On one hand increase in the number of vehicles per year and on the other hand availability of various sensors and communication units at affordable costs encourage to develop a new network called VANET. Nowadays, vehicles are not considered as traditional carriers. Availability of different sensors, front and rear radar, GPS makes the vehicle smart and intelligent [3]. Based on the two components of VANET, the communication in the VANET is classified into two categories known as vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) [4]. Figure 2 shows the components of smart vehicles and the types of communication used in VANET.

The presence of a communication unit in the vehicles and development of radio wireless technology helps to use VANET in a wide range of applications. The application of VANET is classified into four categories. These are safety applications

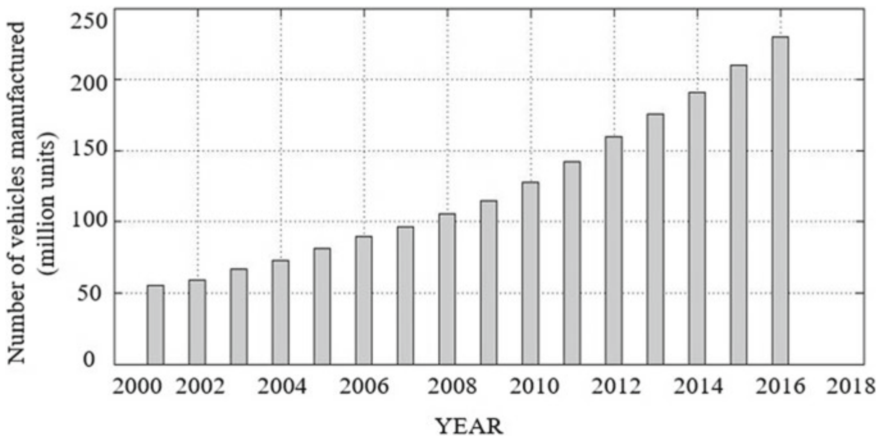


Fig. 1 Number of vehicles manufactured per year

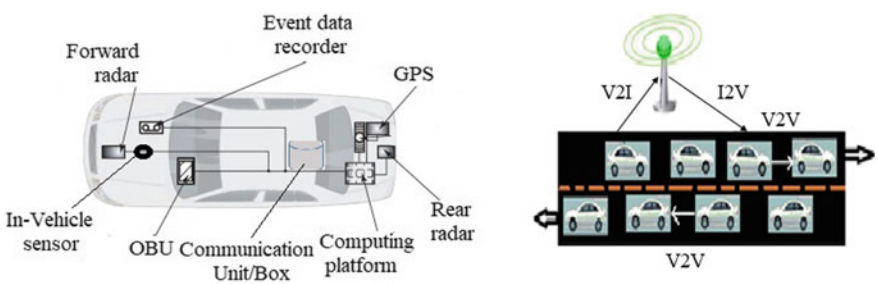


Fig. 2 Components of smart vehicle and types of communication in VANET

[5–7], convenience applications [8, 9], commercial applications [10, 11], and productive applications [12, 13]. Also, the optimization of different generic parameters of VANET motivates to use VANET for a wide range of applications [14].

All these applications will not be successful if either the sensor unit sensing the signal or communication unit receiving and transmitting the signal is the faulty one. This motivates us to design a fault detection algorithm that can detect the fault in OBU to fulfill the objective of VANET for different safety and non-safety applications.

This paper proposed a fault detection mechanism for the faulty OBU of a vehicle. Since the nodes in VANET are performing a lot of computations for the wide range of applications, the computation of fault detection is done at a separate place called a base station. This requires the quick transmission of data from the vehicle to the base station. Vehicular cloud is used for the quick transmission of data through the cloud server of the cloud service provider from the location of the vehicle to the base station. The major contributions of this paper are as follows:

1. A novel fault detection method to detect whether the communication unit (OBU) of the vehicle is faulty or not.
2. Quick transmission of data from the location vehicle to base station through the vehicular cloud.
3. The performance of the fault detection method using Fault Detection Rate (FDR) and Fault Alarm Rate (FAR).

The rest of the paper is organized as follows. Section 2 discusses the literature survey. Section 3 presents the proposed work. Simulation set up and the simulation result is shown in Sect. 4. Finally, the conclusion and future scope are mentioned in Sect. 5.

2 Literature Survey

VANET is one of the current research topics where the researchers are addressing the different issues of VANET. Most of the researchers are focusing on the routing issues of VANET. The objective of routing is to establish a stable route between the source node and the destination node to perform various safety and non-safety application successfully [15]. To address routing, for a different scenario, the routing protocols for VANET are classified into five categories which are shown in Fig. 3.

The number of vehicles is more in city areas. Frequent link disconnection between the nodes in VANET occurs because of random speed and restricted motion of the vehicles due to the structure of the road. For this reason, position-based routing protocols are frequently used [16]. Routing helps in the successful transmission of data from source to destination. But, if the data is faulty then transmission of data by using efficient routing is not going to make the application successful.

The second issue is addressed by many researchers about the security of the transmitted data during transmission through VANET [17]. Different approaches are used to provide security to the data. Blockchain is used to maintain the privacy of

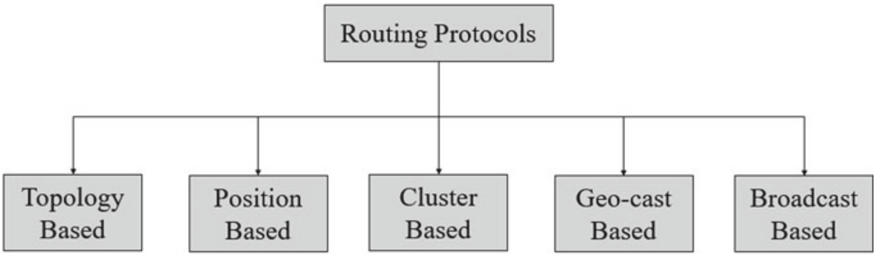


Fig. 3 Types of routing protocol for VANET

data during transmission [18]. To address the different attacks on VANET, different cryptographic approaches are also proposed [19]. Security is really a challenging task and is essential. But the security of faulty data during transmission has no significance. Efficient transmission of data (by efficient routing or through proper security) from one location to another location will not help if the transmitted data is faulty. Faulty data are generated if there is a fault in sensors. Thus, detection of the faulty node is essential to make the application of VANET successful. Different fault detection approaches are proposed in the literature to detect different types of faulty nodes [20–22].

The objective of VANET is to provide faster access to data with lesser delay, to make the network scalable and reliable. To increase the performance of VANET, the computational power, the memory of the vehicle has to be increased. This ultimately affects the cost of the vehicle. For this, many researchers are working to integrate VANET with the cloud to provide mainly three types of services like software as a service, infrastructure as a service, and platform as a service without affecting the cost of the vehicles [23]. For the efficient communication of data, the computational model and the network model is modified with the help vehicular cloud computing model [24]. Thus, this paper determines whether the communication unit of a vehicle is faulty or not, by transmitting the data from the vehicle to the base station through the vehicular cloud and performing computation at the base station.

3 Proposed Work

The proposed work is divided into two phases which are as follows:

1. Transmission phase
2. Fault detection phase.

Assumptions:

Two phases of the proposed work consist of different assumptions which are mentioned as follows.

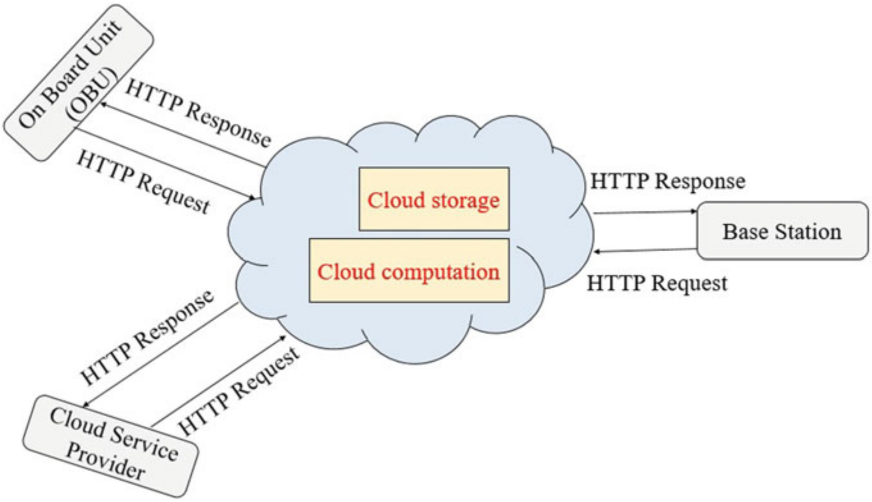


Fig. 4 Transmission phase of proposed work

A1: Since only OBU is checked whether faulty or not, all other sensors of the vehicle are assumed to be fault-free.
A2: Transmission of data to the vehicular cloud is assumed to be fault-free.
A3: All the vehicles and base stations are registered with the cloud to access the cloud service.
A4: No disconnection between the vehicle and the cloud service provider at any instance of time.

1. **Transmission phase**

The objective of this phase is to transmit the data from the vehicle to the base station. For the transmission of data cloud, storage as a service (STaaS) is used. Since the vehicles are registered into the cloud by different cloud service providers, so they can access the storage of the cloud. Since OBU is the communication unit, so the data from OBU is transmitted to cloud storage. Since the fault detection is done at the base station, so the base station accesses the data from the cloud storage. The transmission phase is shown in Fig. 4.

2. **Fault detection phase**

Algorithm 1: Fault detection for the OBU of a vehicle

Algorithm 1 Fault detection algorithm

1: **Input**: Sensed data from cloud
2: **Output**: OBU is faulty or faulty free for vehicle V_i
3: Determine the neg (V_i) of each V_i and calculate D (V_i, neg (V_i))
4: **if** |D(V_i, neg(V_i))| < θ **then**

C (V_i, neg (V_i)) = 0
else
C (V_i, neg (V_i)) = 1
end if
if $\sum_{neg(V_i)} C(V\_i, neg(V\_i)) < \lceil neg\ (V\_i)/2 \rceil$ **then**
V_i = PFF
else
V_i = PF
end if
if $\sum_{neg(V_i)} 1 - 2C(V\_i, neg(V\_i)) \geq \lceil neg\ (V\_i)/2 \rceil$ **then**
V_i = fault free
else
for i = 1 to n **do**
V_i = PFF or PF
if neg (V_i) = fault free ∀ neg (V_i) **then**
if C (neg (V_i), V_i) = 0 **then**
V_i = fault free
else
V_i = faulty
end if
end if
end for
end if
if V_i = fault free **then**
V_i will participate in routing
else
V_i will not participate in communication
end if
STOP

The fault detection phase is carried out at the base station. To determine the fault of the OBU of a vehicle V_i, corresponding neighboring vehicles neg (V_i) has to be determined. For each vehicle V_i, D (V_i, neg (V_i)) is determined where D (V_i, neg (V_i)) is the difference between the sensed data measured by the neighboring vehicle. A comparison test C (V_i, neg (V_i)) is performed by vehicle V_i and its neighbor neg (V_i) using D (V_i, neg (V_i)) and a predefined threshold data θ. Here the value of θ is taken as 10. If D (V_i, neg (V_i)) < θ, then C (V_i, neg (V_i)) = 0, else C (V_i, neg (V_i)) = 1. OBU can be possibly faulty (PF) or possibly fault-free (PFF), decided by performing the comparison test among its neighboring OBU. If $\sum_{neg(V_i)} C(V\_i, neg(V\_i)) < \lceil neg\ (V\_i)/2 \rceil$, then V_i is PFF, otherwise, V_i is PF. V_i is fault-free if it satisfies the following condition.

$$\sum_{neg(V_i)} 1 - 2C(V\_i, neg(V\_i)) \geq \lceil neg(V\_i)/2 \rceil \tag{1}$$

If the V_i is found to be fault-free, then V_i can participate in the routing. Otherwise, no communication through the V_i is carried out. The overall fault detection phase is described in Algorithm 1.

Definition of performance metric
The performance of the fault detection phase is evaluated by two parameters.

1. **False Detection Rate (FDR)**: FDR is the ratio between faulty vehicles detected as faulty to the total number of faulty vehicles present.
2. **False Alarm Rate (FAR)**: FAR is the ratio between the numbers of fault-free vehicles detected as faulty to the total number of fault-free vehicles.

4 Simulation Result and Discussion

For the transmission phase, to store the data ThingSpeak [25] as the cloud storage service provider is used. The data from this cloud storage is retrieved at another location either by JSON, XML, or CSV format. Figure 5 shows the storage of data in the ThingSpeak cloud and the export of data from cloud using CSV format which is used in the fault detection phase.

The performance of the proposed algorithm is evaluated using the MATLAB R2019a simulator. For the fault detection phase, two parameters (FDR and FAR) are computed for the proposed algorithm which is shown in Fig. 6. From the simulation result, it is clear that with an increase in the number of faulty vehicles, FAR increases and FDR decreases. Also, greater is the density of the vehicles in the network, greater is the FAR, and lesser is the FDR.

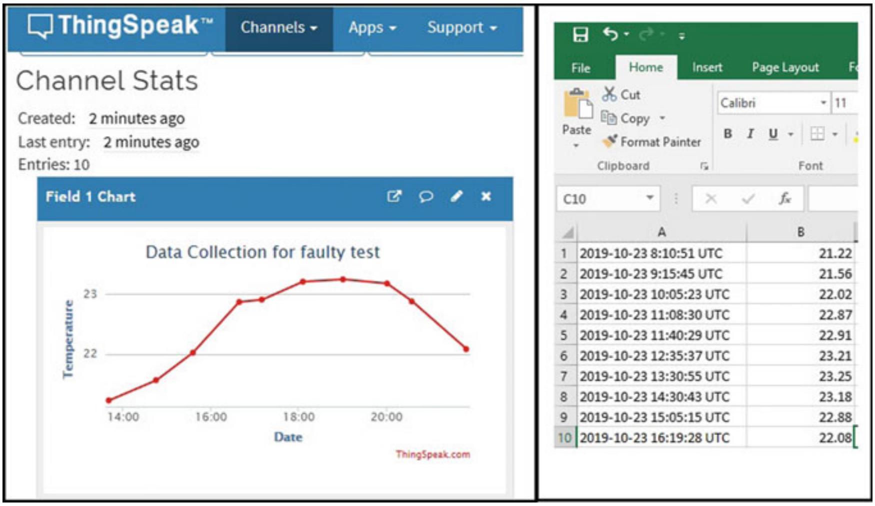


Fig. 5 Data stored in ThingSpeak cloud and its extraction in CSV format

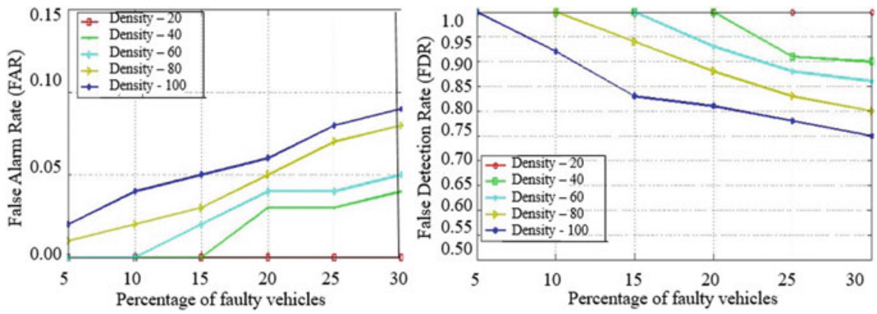


Fig. 6 Variation of FAR and FDR w.r.t. percentage of faulty vehicles

5 Conclusion and Future Scope

In this paper, the OBU of the vehicle is determined as faulty or fault-free. On the basis of the fault detection result, the vehicle either participates in the routing otherwise the vehicle is isolated without any communication. The proposed algorithm gives a high detection rate and low false alarm rate. In the future, we would compare the proposed work with other fault detection approaches and would determine other types of fault.

References

1. Bhoi, S.K., Khilar, P.M.: Vehicular communication: a survey. IET Netw. **3**(3), 204–217 (2013)
2. https://www.statista.com/statistics/281134/number-of-vehicles-in-useworldwide/
3. Abdelhamid, S., Hassanein, H., Takahara, G.: Vehicle as a resource (VaaR). IEEE Netw. **29**(1), 12–17 (2015)
4. Cunha, F., Villas, L., Boukerche, A., Maia, G., Viana, A., Mini, R.A.F., Loureiro, A.A.F.: Data communication in VANETs: protocols, applications and challenges. Ad Hoc Netw. **44**, 90–103 (2016)
5. Khekare, G.S., Sakhare, A.V.: A smart city framework for intelligent traffic system using VANET. In: 2013 International Multi-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), pp. 302–305. IEEE (2013)
6. Oliveira, R., Montez, C., Boukerche, A., Wangham, M.S.: Reliable data dissemination protocol for VANET traffic safety applications. Ad Hoc Netw. **63**, 30–44 (2017)
7. Younes, M.B., Boukerche, A.: Efficient traffic congestion detection protocol for next-generation VANETs. In: 2013 IEEE International Conference on Communications (ICC), pp. 3764–3768. IEEE (2013)
8. Senapati, B.R., Khilar, P.M., Sabat, N.K.: An automated toll gate system using VANET. In: 2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP), pp. 1–5. IEEE (2019)
9. Jayapal, C., Roy, S.S.: Road traffic congestion management using VANET. In: 2016 International Conference on Advances in Human Machine Interaction (HMI), pp. 1–7. IEEE (2016)
10. Bhoi, S.K., Puthal, D., Khilar, P.M., Rodrigues, J.J.P.C., Panda, S.K., Yang, L.T.: Adaptive routing protocol for urban vehicular networks to support sellers and buyers on wheels. Comput. Netw. **142**, 168–178 (2018)

11. Zhang, Y., Zhao, J., Cao, G.: Roadcast: a popularity aware content sharing scheme in VANETs. ACM SIGMOBILE Mob. Comput. Commun. Rev. **13**(4), 1–14 (2010)
12. Senapati, B.R., Swain, R.R., Khilar, P.M.: Environmental monitoring under uncertainty using smart vehicular ad hoc network. In: Smart Intelligent Computing and Applications, pp. 229–238. Springer, Singapore (2020)
13. Safi, Q.G.K., Luo, S., Pan, L., Liu, W., Yan, G.: Secure authentication framework for cloud-based toll payment message dissemination over ubiquitous VANETs. Pervasive Mob. Comput. **48**, 43–58 (2018)
14. Senapati, B.R., Khilar, P.M.: Optimization of performance parameter for vehicular ad-hoc network (VANET) using swarm intelligence. In: Nature Inspired Computing for Data Science, pp. 83–107. Springer, Cham (2020)
15. Dutta, R., Thalore, R.: A review of various routing protocols in VANET. Int. J. Adv. Eng. Res. Sci. **4**(4) (2017)
16. Liu, J., Wan, J., Wang, Q., Deng, P., Zhou, K., Qiao, Y.: A survey on position-based routing for vehicular ad hoc networks. Telecommun. Syst. **62**(1), 15–30 (2016)
17. Rivas, D.A., Barceló-Ordinas, J.M., Zapata, M.G., Morillo-Pozo, J.D.: Security on VANETs: privacy, misbehaving nodes, false information and secure data aggregation. J. Netw. Comput. Appl. **34**(6), 1942–1955 (2011)
18. Lu, Z., Liu, W., Wang, Q., Qu, G., Liu, Z.: A privacy-preserving trust model based on blockchain for VANETs. IEEE Access **6**, 45655–45664 (2018)
19. Mejri, M.N., Ben-Othman, J., Hamdi, M.: Survey on VANET security challenges and possible cryptographic solutions. Veh. Commun. **1**(2), 53–66 (2014)
20. Mohapatra, S., Khilar, P.M.: Artificial immune system based fault diagnosis in large wireless sensor network topology. In: TENCON 2017–2017 IEEE Region 10 Conference, pp. 2687–2692. IEEE (2017)
21. Mohapatra, S., Khilar, P.M.: Forest fire monitoring and detection of faulty nodes using wireless sensor network. In: 2016 IEEE Region 10 Conference (TENCON), pp. 3232–3236. IEEE (2016)
22. Mohapatra, S., Khilar, P.M., Swain, R.R.: Fault diagnosis in wireless sensor network using clonal selection principle and probabilistic neural network approach. Int. J. Commun. Syst. e4138 (2019)
23. Bitam, S., Mellouk, A., Zeadally, S.: VANET-cloud: a generic cloud computing model for vehicular ad hoc networks. IEEE Wirel. Commun. **22**(1), 96–102 (2015)
24. Lee, E., Lee, E.-K., Gerla, M., Oh, S.Y.: Vehicular cloud networking: architecture and design principles. IEEE Commun. Mag. **52**(2), 148–155 (2014)
25. https://thingspeak.com/

An Approach to Cohort Selection in Cloud for Face Recognition

Jogendra Garain, Satyabrata Maity, and Dipak Kumar

Abstract Face recognition is such a problem where the accuracy is dependent on the quantity and variations of the training images. Increase of training data enhances the recognition accuracy in most cases. However, to collect and store a huge number of data in a standalone storage device is a big problem. Therefore, the researchers are being motivated towards cloud computing. Nowadays, the technology of cloud computing has made the task fairly ease even when a huge volume of data is required. This paper presents a cloud-based cohort selection approach which can be incorporated into a face recognition system. In this scheme, the cohort images as well as the training images can be accessed from the cloud servers through network connectivity. So the storage and maintenance costs are reduced whereas the accuracy is increased.

Keywords Face recognition · Cohort selection · Cloud computing · Cohort images

1 Introduction

Due to the high demand for user's authenticity in the modern digital world, face recognition has a lot of space in research. Although extensive progress has been achieved in this area, yet it sometimes suffers from a few factors. Insufficient image samples are one of them. Cohort selection in face recognition has resolved the issue to some extent with the help of non-matched images. The general meaning of the term 'cohort' is a set of people that carries a number of similar characteristics.

J. Garain (✉) · S. Maity
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: jogendragarain@soa.ac.in

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Kumar
Department of Computer Science and Engineering, Government Polytechnic Sitamarhi, Birpur, Sursand, Sitamarhi, Bihar, India
e-mail: dipakcsi@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_11

Cohort was first introduced by Frost in 1935 [1]. In the present research trend of face recognition, huge numbers of training images are used because a FR system usually becomes more robust, if it is trained with more image samples. However, it is not at all expected to store all the images in a local server because it will drastically increase the overhead of storing and maintaining costs. This paper presents a framework to make a face recognition system more accessible to users by the virtue of cloud computing technology. The construction of cohort pool needs a collection of images with sufficient intra-class variations. To store all those images in local storage is a big overhead. Therefore, it is pretty attractive to use images from different cloud servers for preparing the cohort pool as well as the training set.

Three different kinds of models are there for cloud computing services like Platform as a Service (PaaS), Software as a Service (SaaS) and Infrastructure as a Service (IaaS). The advantages of accessing the images from cloud storage are: it reduces the overhead of storing and maintaining the data; it increases the variations among inter-class images since the images can be accessed from different cloud servers and it also increases the recognition accuracy. However, the user should concern about the privacy of the images which are stored in the cloud and are accessed through the internet. Guo et al. [2] have proposed a method for preserving the privacy of the information used for face recognition even when they are stored in the cloud.

The paper is structured as follows. Related literature survey has been discussed in Sect. 2. The proposed architecture has been depicted and explained in Sect. 3. The cohort selection mechanism is stated in Sect. 4. The database and results are discussed in Sect. 5. Section 6 concludes.

2 Literature Survey

Haghighat et al. [3] proposed a cloud-based identification system with security concerns. The authors present CloudID, a privacy-preserving cloud-based and cross-enterprise biometric identification solution. The authors have applied k-d tree structure in the core of the searchable encryption. In this literature, they claim that this is the first cloud-based biometric identification system. Siregar [4] proposes a face recognition system that uses a cloud computing mechanism. The authors in this work have used eigenface to represent a face image and REST to collect the data from clouds. Von Söhsten and Murilo [5] proposed a real-time face recognition using cloud computing, EMGU CV and Triple Modular Redundancy. The authors have shown a hike in the performance of the face recognition system. Wide use of mobile phones has resolved some crucial problems, especially the issues of importability. However, a mobile device has low storage. The computing power of them is generally not very strong. These issues are solved to some extent by the application of cloud computing or Mobile Cloud Computing (MCC) [6]. An extensive survey on cloud computing research trends and publications on this domain are presented by Heilig and Vob in [7]. Stojmenovic states about the benefits of mobile cloud computing in

[8]. Buyya et al. [9] discusses the utilities of cloud computing in the present market for business purposes.

A cloud-based face recognition system's infrastructure has been proposed by Kisku and Rana [10] in which the authors have used PCA and SIFT features to characterize the face images. They have computed six PCA-featured instances of each face image. To enhance the performance, they have incorporated cohort selection mechanism with their proposed face recognition system. A hybrid heuristic cohort selection is proposed in their work.

3 Architecture of the Proposed Approach

The cohort images are basically the non-match templates. They can be from the gallery images or other than the gallery images. In the first choice, the images must be locally stored in the system's storage in which the process of recognition is conducted.

However, for the non-gallery cohort images, they can be stored either in the local storage device or in the cloud storage. The following architecture is a proposed approach to use cloud computing technology for cohort selection. It reduces the extra overhead of storing and maintaining the non-match templates for cohort selection. The framework is shown in Fig. 1 presents a face verification system where both the query image and claimed id are provided as input. The claimed id must be one of the enrolled ids, otherwise, the subject has to be registered first. After the required preprocessing, the query image is matched with enrolled images that are stored locally where the verification is performed. This matching provides a genuine score (g). However, the cohort images are stored in the cloud server. The cloud engine already executes the cohort selection process to select a specific cohort subset for all enrolled subjects before the verification process starts. When the probe image claims an identity then this information is sent to the cloud engine to retrieve the corresponding set of cohort images (as shown in Fig. 1) which is matched with the probe image and generates a set of matching scores, called as cohort scores (δ). After that, the genuine score 'g' is normalized with the help of cohort score's set using cohort score normalization techniques [11]. The normalized scores are compared with the threshold value and the final decision of acceptance is governed accordingly.

4 Cohort Selection Mechanism

In this paper, the Bezier Curve Cohort Selection (BCCS) algorithm [12] has been explained. However, any robust cohort selection algorithm can be used in the proposed architecture, shown in Fig. 1. The initial cohort pool is constructed by the cloud engine with the images stored in its storage. Then the matching scores are calculated. All the matching scores obtained from the non-matched templates, called cohort scores, are considered as control points (c) of a Bezier curve (B) and then

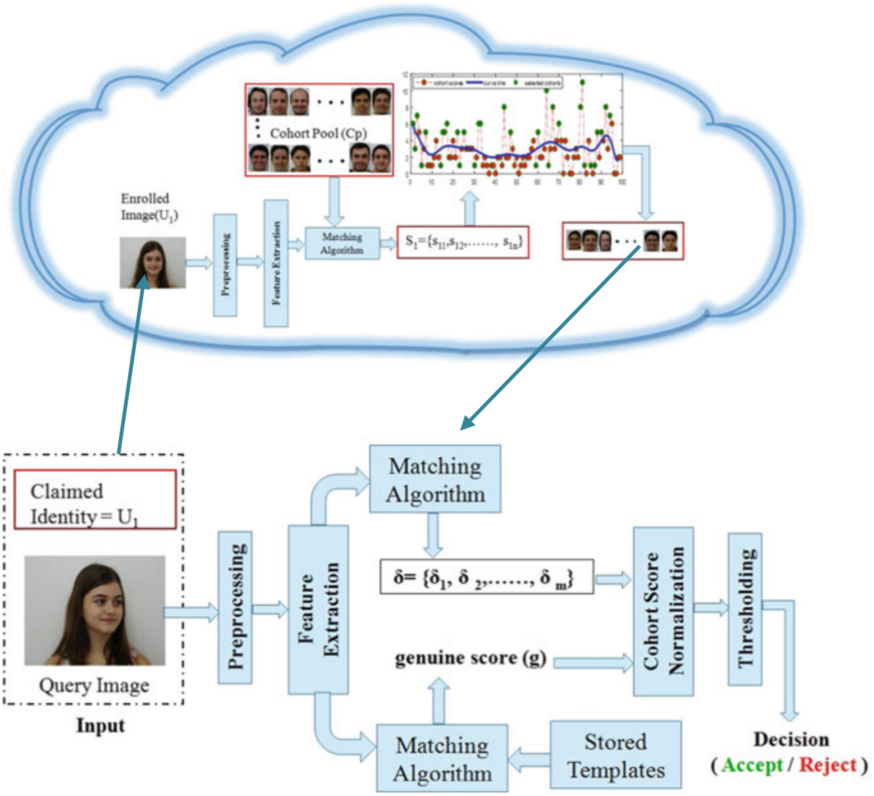


Fig. 1 Architecture of the proposed approach: cohort selection in cloud for face recognition

the curve is drawn using De Casteljau algorithm [13] as shown in Eq. (1) where the degree of Bezier curve is one less than the number control points. If the number of cohort templates in the initial cohort is n then the number of cohort scores is also n (If no self instances present in the pool) Therefore, the degree of the Bezier curve consisting of the cohort scores obtained from the initial cohort pool of length n will be of $n - 1$. The fact is represented in Eq. (1).

$$B(t) = \sum_{i=0}^{n-1} b_i^n(t) \cdot c_{i+1}, \quad 0 \le t \le 1 \tag{1}$$

where $b_i^n = (i^n) \cdot t^i \cdot (1-t)^{n-1}$ is the n degree Bernstein polynomial.

After that, the distance (d_i) from each control points (c_i) to the curve (B) are calculated and mean (μ) of all these distances are determined. Finally, the control points are selected for the cohort subset as per Eq. (2).

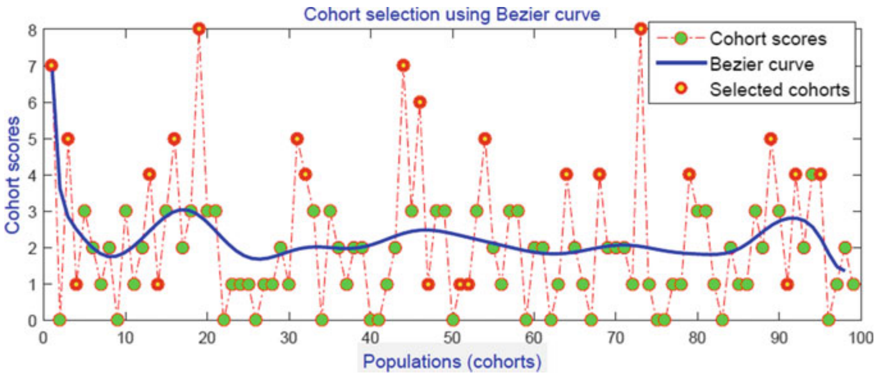


Fig. 2 Cohort selection using BCCS algorithm

$$T_i \in C_S \text{ if } d_i \geq \mu \tag{2}$$

where T_i is the corresponding template of ith cohort score, C_S is the selected cohort subset for the ith subject. Figure 2 shows a sample of cohort selection using BCCS algorithm on the FEI face database [14].

To find out the matching scores, we need to extract the facial features from each face images available in the initial cohort pool. We have used the SURF points [15] in the proposed approach to encode the face images.

4.1 Feature Extraction

SURF [15] is a feature, invariant to scaling, rotation and illumination (partially). Therefore, it helps to deal with various challenges available in the images. A template is generated for each face image using SURF features and it is stored either in the same cloud storage or different cloud storage as per the availability. The localization of SURF points on the face images is shown in Fig. 3.

4.2 Finding of Match Score

The matching score for each pair of images is calculated using the Best-fit matching algorithm [16]. This algorithm works on dynamic threshold, so it reduces number of false matches. The scores are further applied for cohort selection. A sample of matching between two face images (different instances) of same person is shown in Fig. 4.

Fig. 3 Showing the SURF points on face images

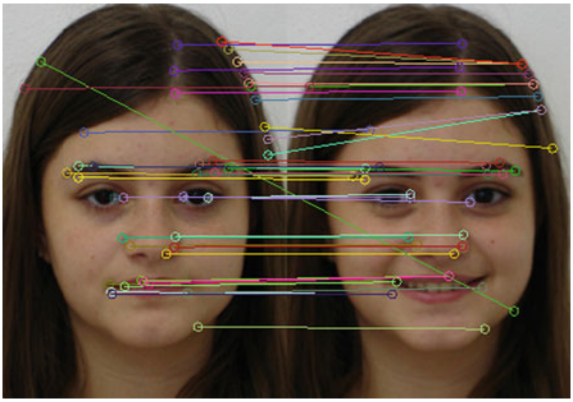


Fig. 4 SURF point matching

5 Database and Results

The FEI face database is used to conduct the experiments. Samples are shown in Fig. 5. This database consists of 2800 face images with the variations of expression (smiling), rotation and illumination. There are total 200 subjects; each of them is having 14 samples. Two hundred smiling images are taken into the cohort pool and BCCS algorithm finds the suitable cohorts from that pool. Figure 6 shows the number of cohorts selected for each person from the FEI database.

6 Conclusion

The cohort images may improve the accuracy of a face recognition system. However, to store a separate set of cohort images needs extra storage and it increases the maintenance overhead too. Therefore, use of cloud technology may give relief from

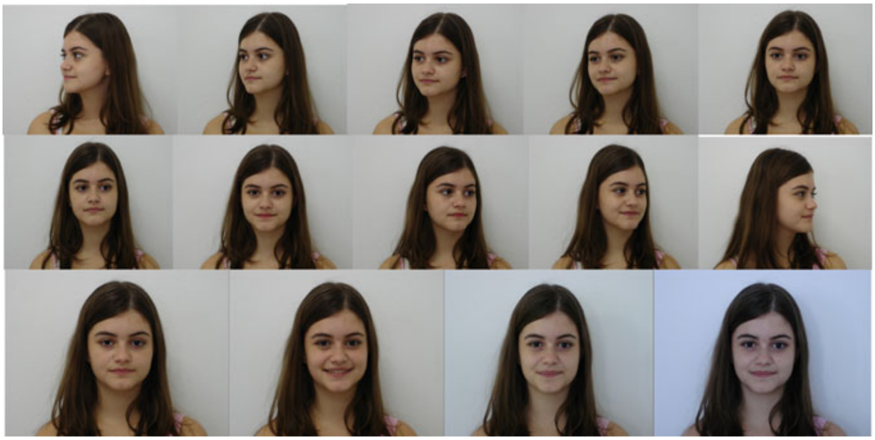


Fig. 5 Image samples from FEI database

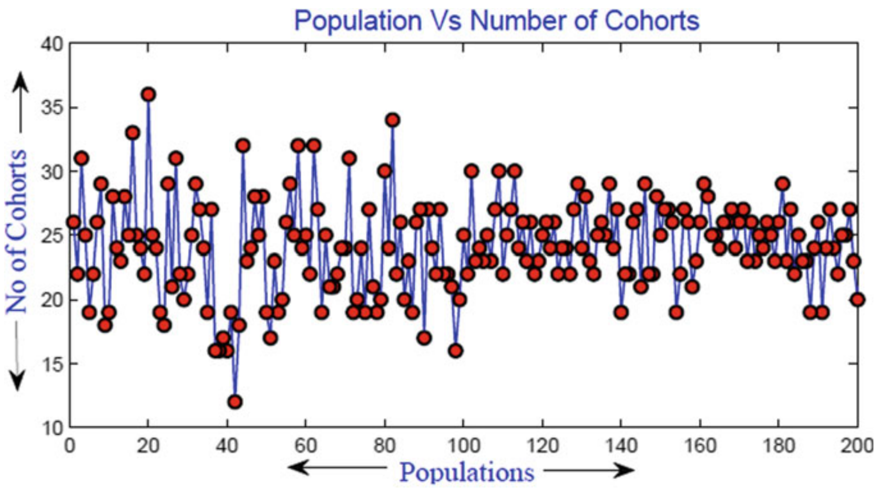


Fig. 6 Number of selected cohorts for each subject

those problems to some extent. This paper proposes an approach of cohort selection using cloud computing techniques where the extra cost of storage and maintenance is reduced. The proposed technique is not limited to the local features only as SURF has been used for feature extraction. The other facial features (local as well as global) are also applicable. The security of the cohort images stored in the cloud is also an important issue that has not been included in this paper. So it can be a future scope to extend the proposed approach by incorporating the security aspects.

References

1. Frost, W.H.: The age selection of mortality from tuberculosis in successive decades. Am. J. Epidemiol. **30**(3), 91–96 (1939)
2. Guo, S., Xiang, T., Li, X.: Towards efficient privacy-preserving face recognition in the cloud. Signal Process. (2019)
3. Haghighat, M., Zonouz, S., Abdel-Mottaleb, M.: CloudID: trustworthy cloud-based and cross-enterprise biometric identification. Expert Syst. Appl. **42**(21), 7905–7916 (2015)
4. Siregar, S.T.M., Syahputra, M.F., Rahmat, R.F.: Human face recognition using eigenface in cloud computing environment. IOP Conf. Ser. Mater. Sci. Eng. **308**(1), 012013 (2018)
5. von Söhsten, D., Murilo, S.: Multiple face recognition in real-time using cloud computing, Emgu CV and triple modular redundancy. J. Inf. Assur. Secur. **9**(4) (2014)
6. Ayad, M., Taher, M., Salem, A.: Real-time mobile cloud computing: a case study in face recognition. In: 2014 28th International Conference on Advanced Information Networking and Applications Workshops, pp. 73–78. IEEE (2014)
7. Heilig, L., Vob, S.: A scientometric analysis of cloud computing literature. IEEE Trans. Cloud Comput. **2**(3), 266–278 (2014)
8. Stojmenovic, M.: Mobile cloud computing for biometric applications. In: Proceedings of the 15th International Conference on Network-Based Information Systems (NBIS'12), pp. 654–659 (2012)
9. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: 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 (2009)
10. Kisku, D.R., Rana, S.: Multithread face recognition in cloud. J. Sens. **2016** (2016)
11. Merati, A., Poh, N., Kittler, J.: User-specific cohort selection and score normalization for biometric systems. IEEE Trans. Inf. Forensics Secur. **7**(4) (2012)
12. Garain, J., Kumar, R.K., Kisku, D.R., Sanyal, G.: Selection of user-dependent cohorts using Bezier curve for person identification. In: International Conference Image Analysis and Recognition, pp. 566–572 (2016)
13. Sir, Z., Juttler, B.: On de Casteljau-type algorithms for rational Bezier curves. J. Comput. Appl. Math. **288**, 244–250 (2015)
14. Tenorio, E.Z., Thomaz, C.E.: Analise multilinear discriminante de formas frontais de imagens 2D de face. In: Proceedings of the X Simposio Brasileiro de Automacao Inteligente SBAI 2011, pp. 266–271. Universidade Federal de Sao Joao del Rei, Sao Joao del Rei, Minas Gerais, Brazil (2011)
15. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: speeded up robust features. In: European Conference on Computer Vision, pp. 404–417 (2006)
16. Garain, J., Shah, A., Kumar, R.K., Kisku, D.R., Sanyal, G.: BCP-BCS: best-fit cascaded matching paradigm with cohort selection using Bezier curve for individual recognition. In: Asian Conference on Computer Vision, pp. 377–390 (2016)

Internet of Things (IoT) Framework Deployment Template for Cloud-Based Harbor Surveillance and Ferry Monitoring System

Madhuri Rao, Narendra Kumar Kamila, and Nibedita Adhikari

Abstract Cloud computing is a lucrative option to deal with data that is huge with high velocity; the kind that is generated by sensors installed in harbor monitoring and surveillance. Real-time data is collected by sensors deployed under the sea surface and along the coast. Sensors installed on ships enable monitoring ship movements and facilitate intrusion detection. These sensors generate huge volume of data which is not feasible to be stored locally given the hardware constrains of the sensor devices. Such data can be assimilated and forwarded via a gateway to a cloud storage space. Data and processed information on cloud can be accessed with the help of customized cloud applications by ferry passengers, ferry owners, and harbor administration staff as availed cloud services. This paper presents a deployment template of a cloud-based IoT framework for harbor surveillance and ferry monitoring system. Here, the functional metrics such as average service time is determined with the help of stochastic queuing system.

Keywords Cloud-based storage · Internet of things (IoT) · Harbor surveillance · Ferry monitoring · Deployment template · Queuing system

1 Introduction

Today, we live in a world surrounded by sensors which make our cities smarter. It is predicted [1] that by 2050, about 66% of the world population will thrive in cities. Therefore, our cities have to invest in techniques that enable smart administration and management of resources. Many cities are being re-planned with ferry transport as an alternate to terrestrial and air transport. Kochi Water Metro [2] is one such

M. Rao (✉) · N. Adhikari
Biju Patnaik University of Technology, Rourkela, Odisha 769008, India
e-mail: madhurirao.iter.soa@gmail.com

N. K. Kamila
BRM IIT, Bhubaneswar, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_12

project that is being built by the State Government of Kerala, India. Modernized air-conditioned ferry equipped with Wi-Fi is proposed to ply over seven routes with a capacity to accommodate 50–100 passengers. Three types of users may be interested in tracking information related to arrival, departure, schedule, and availability of seats. On September 15, 2019, around 12 people died and 30 went missing when one such ferry got capsized in the swollen river Godavari, Andhra Pradesh, India [3]. Cities in India with rivers are usually densely populated and require water as an alternate means of commutation, as road transport is already congested. Here, harbor has to cater systems that can be better monitored. This shall immensely help in traffic management and adequately equip information for better safety and rescue operations.

2 Proposed System

Sensors are being embedded everywhere. Sensors help in collecting information from regions that are dynamic and often not easily accessible. Information collected by sensors, when assimilated help in making better decisions. They indeed, enhance the efficiency of actuation systems. They are the building blocks of the Internet of things (IoT) technology [4]. Have presented how sensors could be used in tracking ship movements. In [5], a group of sensors are incorporated for building efficient harbor surveillance systems [6]. Present a novel tracking technique for detecting intruders approaching a sailing ship. Currently, ship traffic is monitored with the help of satellite images. Ship traffic within a limited range can be monitored more efficiently with the help of wireless sensors. This is possible by integrating cloud technology and a mobile application with the information of ship traffic collected by a customized infrastructure of sensors. A simple mobile application can be developed for end users, who require commuting by ferry regularly. This mobile application can be hosted on a public or private cloud, as per the policies of the ferry transport provider company. Cloud enhances the efficiency of mobile computing and creates a computing environment that can substantially reduce the cost of operation. Connecting to a server on cloud enhances the capabilities and facilities of a program that traditionally only computed some basic metrics. If information collected from sensors is connected to a cloud, then some of the benefits of cloud computing can also be explored. Sensor network deployed on a harbor and along coast line could be connected to a cloud infrastructure with the help of a gateway node that provides controller services. Sensors deployed on ship enable tracking ship motions which could be heave, sway, surge, yaw, pitch, and roll. Here, pressure and position sensors are required to collect information of its surrounding in every few seconds. This creates a lot of information that cannot be stored in sensors as they have limited memory. This information can be forwarded via a gateway to a device for long-term storage. The information stored in this storage device can further be processed for controlling and monitoring traffic. If this is hosted on a cloud server, end users such as ferry passengers and ship owners can tap to get customized information via a

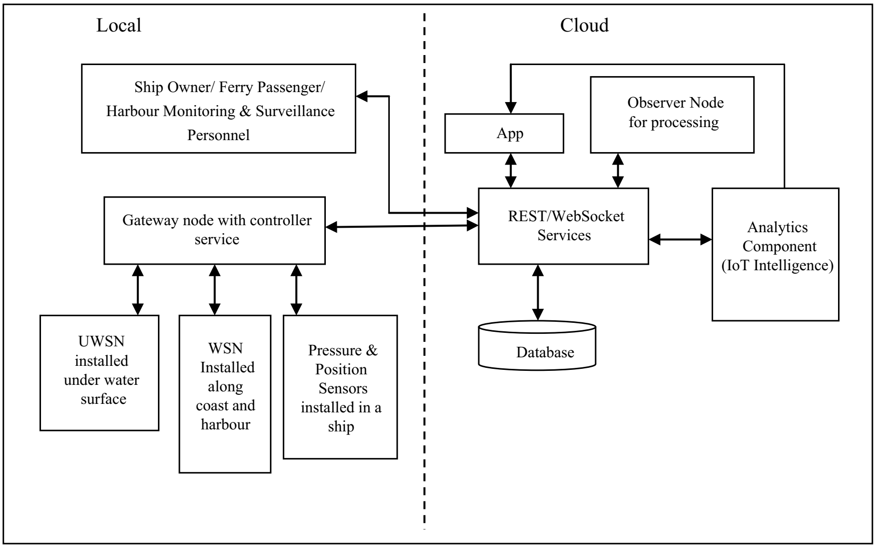


Fig. 1 IoT deployment template for harbor monitoring and ship motion surveillance system

mobile application. Android being an open source facilitates applications to reach across a spectrum of such applications from mobile device. Information collected by sensors deployed along coast and harbor can help in predicting and controlling ferry traffic. The system is perceived with a three-layer approach. Sensor centric layer is responsible for creation and maintenance of sensor network with an aim to ensure that data is seamlessly connected and forwarded to the virtual sensor of the middleware layer. The virtual sensors are essentially coordinator nodes that facilitate controller services. They act like gateways connecting the sensor network to the cloud with the help of Representational State Transfer (REST) or Web socket communication application interfaces. The middleware is also responsible for provisioning and billing of resources as desired from the cloud. The client-centric layer manages sessions of end users that connect to the cloud through customized applications. Figure 1 depicts the proposed IoT deployment template for cloud-based harbor surveillance and ship motion monitoring system.

In the next section, the proposed model is simulated as a queuing system.

3 Model Building

Three entities are considered here, namely ferry passengers, ferry owners, and harbor administration staff as depicted is the process flow chart given here as Fig. 2. The queuing system is considered with service request of entities as a Poisson arrival process. The model as depicted in Fig. 1 is ideally M/G/m/m queuing system, where

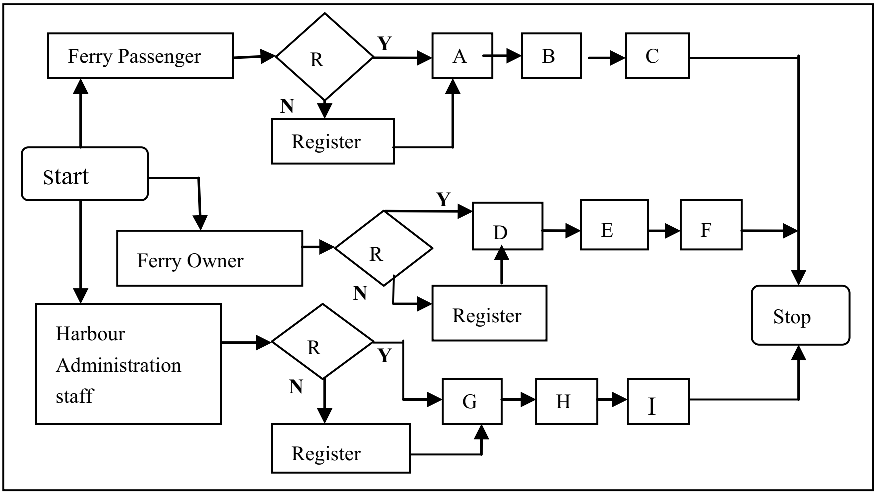


Fig. 2 Process flowchart harbor monitoring and ship motion surveillance system (R—registration validation service, A—ferry schedule, departures, B—ferry seat availability and selection service, C—ferry seat billing and booking confirmation service; D—ferry security monitoring service, E—ferry passenger details and raised billing information service, F—ship tracking service; G—harbors security alert service, H—ferry docking details, I—ferry schedules and passengers details)

there are multiple servers to serve multiple entities with multiple services and the service time of the servers is assumed to have a general distribution. Such queue supports easy analyses and therefore is often considered more suitable for IoT-based applications. The m servers servicing the various requests of the various entities are part of the cloud infrastructure and ideally have no waiting time for serving the customers. The three entities are first required to be registered users of the system. After initial validation, the registered users can avail the services through their application. Ferry passenger entity first has to go through the registration validation process. If a user is not registered, he or she will be guided through the registration process. Once a user is registered, he or she is allowed to avail services A, B, and C. Service A facilitates checking of ferry schedule and departures, Service B allows users to check for seat availability and select seats if available. Service C allows users to pay for the seat selected and finally receive confirmation of the booked ferry ticket.

Ferry owner is the second entity of the proposed model, which initially has to go through registration validation process. If the ferry owner is not a registered user, he or she will be guided to the registration process. A registered ferry owner, once validated can avail the further services D, E, and F. Service D enables the ferry owner to monitor his ferry security measures by accessing streaming surveillance video and Wi-Fi data that is being accessed through the ferry's Wi-Fi router.

Additionally, they would need information that caters to security and day-to-day functions of the Harbor Management and Maintenance. A registered harbor staff can avail services G, H, and I through customized and secured applications that connect

Table 1 Simulation details of the queuing system solution for the ferry passenger entity

Process in queue	Time between arrivals [min]	Probability	Mapping
Registration process and validation	5	0.5	0–49
	10	0.3	50–79
	15	0.2	80–99
A	5	0.5	0–49
	10	0.3	50–79
	15	0.2	80–99
B	10	0.7	0–69
	5	0.2	70–89
	1	0.1	90–99
C	10	0.5	0–49
	5	0.5	50–99

to could services. Service G shall enable the staff in getting security alerts, while service H essentially provides data about the currently docked and sailing ships. Service I shall enable entity harbor staff to get information about the ferry schedules and details of passengers boarding and landing. As this is a work in progress, only the ferry passenger entity is simulated in queuing system. The other entities will be simulated in our future work. Table 1 provides the simulation details of queueing system considered for ferry passenger entity. From Table 1, the average inter-arrival time for users to the system can be computed as 1/λ, where λ is the arrival rate.

As per Little's Rule [7]

$$L = \lambda W;\ \ L\mathrm{q} = \lambda W\mathrm{q} \tag{1}$$

where Wq denotes mean waiting time in the queue.

By intuition, it is inferred that

$$W = W\mathrm{q} + (1/\mu) \tag{2}$$

Here, μ denotes mean service rate. Equation (2) implies that the mean wait in the system is the sum of mean wait time in the queue and the service time $(1/\mu)$ (Fig. 3).

Hence, using the time between arrivals of service request of ferry passengers based on probability values mapped to arrival of processes with Poisson distribution as depicted in Table 1, the average inter-arrival time can be computed as $1/\lambda = 0.5 * (5) + 0.3 * (10) + 0.2 * (15) = 8.5$ min. Further, average rate $\lambda = 1$/average inter-arrival time $= 0.1176$ services/min $= 7.06$ services/h.

The average service time of Service at Process A can similarly be computed from Table 1 as $1/\mu\mathrm{A} = 0.5 * (5) + 0.3 * (10) + 0.2 * (15) = 8.5$ min. The average service time of Service at Process B $= 1/\mu\mathrm{B} = 0.7 * (10) + 0.2 * (5) + 0.1 * (1) = 8.1$ min. Similarly, the average service time of Service at Process C $= 1/\mu\mathrm{C} = 0.5 * (10) +$

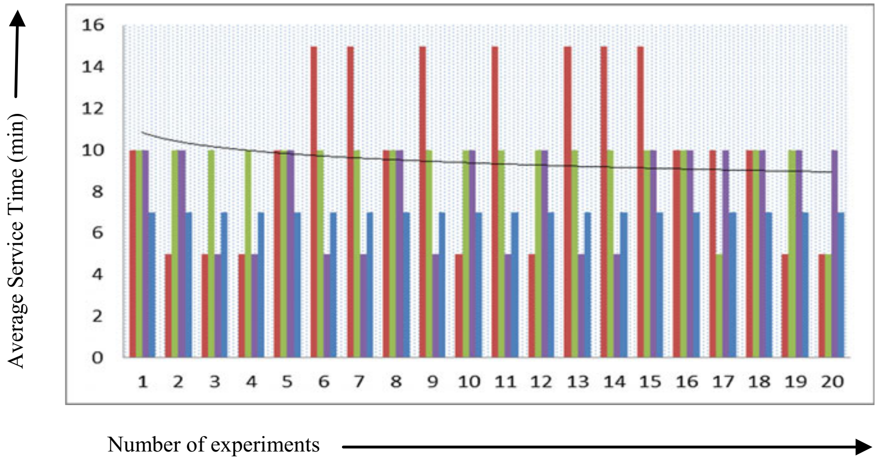


Fig. 3 Graphical representation of average service time of registration process, Process A, Process B, and Process C for ferry passenger entity

$0.5 * (5) = 7.5$ min. We consider a constant service time of 7 min for the registration process.

Using Eq. (2), the total time spent by the ferry passenger entity in availing the services is hence

$$T = W_{\text{Registration}} + W_{\text{A}} + W_{\text{B}} + W_{\text{C}} + 1/\mu_{\text{registration}} + 1/\mu_{\text{A}} + 1/\mu_{\text{B}} + 1/\mu_{\text{C}} \quad (3)$$

3.1 Data Set and Test

Data set is synthesized with the help of probability axioms and random distribution functions. Ghaleb et al. [8] have used Arena Software to model and compute average time spent by students in a college restaurant. As data cannot be collected in our case, we synthesize data randomly for service time of each process, namely Registration process, Process A, Process B, and C. We generate random values between 0 and 100 by using RAND function and generate random service time by assuming certain probable values with respect to weights considered based on considered distribution and the value of the random value. Table 1 summarizes the simulation details of the entity ferry passenger, where we assume that all four sub-processes are also random processes.

The model was tested in Microsoft Excel ver 2010 and was run for 20 replications. Figure 4 depicts the graphical representation of the average inter-arrival time of ferry passenger entity, where Y-axis is the time in min and X-axis represents number of ferry passengers.

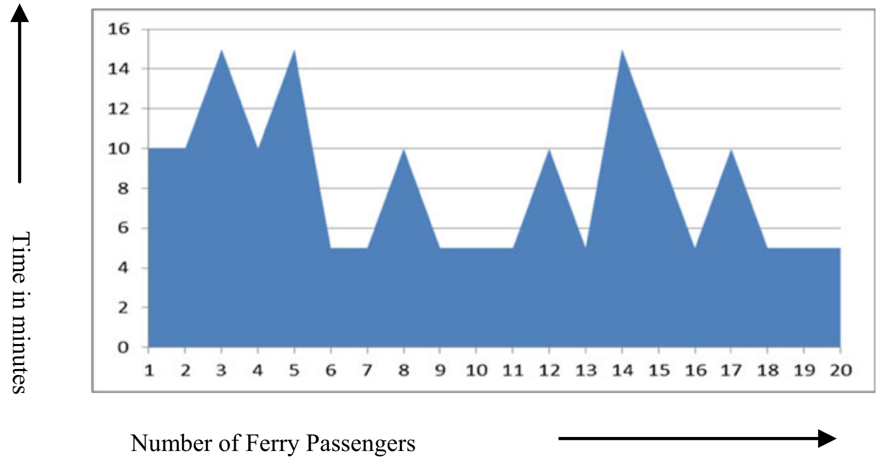


Fig. 4 Graphical representation of inter-arrival time (min)

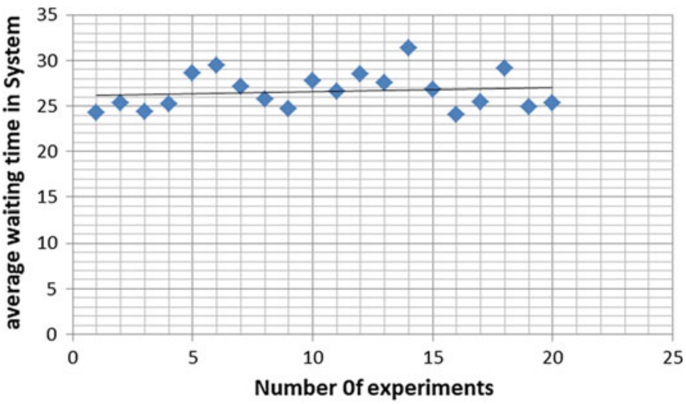


Fig. 5 Average time spent (min) by the ferry passenger in system

Figure 5 depicts the graphical representation of the average time spent in min by ferry passenger entity. Here, Y-axis is the time in min and X-axis represents number of times the simulation was run in order to see the performance of the model.

4 Discussions

It can be inferred that the proposed queuing model approach is suitable in predicting and determining the average time spent by the entity ferry passenger in availing resources from the cloud platform. This helps in designing an efficient deployment template for IoT systems. Queuing theory can help in analyzing how resources are utilized and how idle or congested a path to resource is. As this is a work in progress, we are unable to comment on the overall system constraints.

5 Conclusion

The harbor surveillance and ferry monitoring system are IoT applications that generate a huge amount of data by sensors deployed at various levels. This huge quantity of data can ideally be mounted on a cloud. Deployment of such an IoT framework is discussed here. The system is characterized by the types of users who will access the cloud for availing customized services. Further a part of the proposed sub-system is modelled using queuing theory to determine the average service time of the ferry passenger entity.

References

1. United Nation: World Urbanization Prospect. [Online]. Available: http://dl.acm.org/citation.cfm?id=308574.308676 (2014)
2. Kochi Water Metro project moves ahead in full steam. Manorama Online, 18 June 2016. Retrieved 22 July 2016
3. https://economictimes.indiatimes.com/news/politics-and-nation/several-people-feared-drowned-in-tourist-boat-capsize-in-andhra-pradesh/articleshow/71136283.cms?from=mdr
4. Rao, M., Kamila, N.K.K.: Bayesian network based energy efficient ship motion monitoring. Karbala Int. J. Mod. Sci. **4**, 69–85 (2018)
5. Rao, M., Kamila, N.K., Kumar, K.V.: Underwater wireless sensor network for tracking ships approaching harbor. In: International Conference on Signal Processing, Communication, Power and Embedded System, Paralekhmundi, India (2016)
6. Rao, M., Kamila, N.K.: Tracking intruder ship in wireless environment. Hum. Centric Comput. Inf. Sci. **7**, 1–26 (2017)
7. El-Taha, M., Stidham, S.: Little's formula and extensions. In: Sample-Path Analysis of Queueing Systems. International Series in Operations Research & Management Science, vol. 11. Springer, Boston, MA (1999)
8. Ghaleb, M.A., Suryahatmaja, U.S., Alharkan, I.M.: Modelling and queuing systems using arena software: a case study. In: International Conference on Industrial Engineering and Operations Management, Dubai, UAE, 3–5 Mar 2016

Art of Style Transfer Using Convolutional Neural Network: A Deep Learning Approach

Siddharth Saurabh and Barnali Sahu

Abstract In the present scenario, many difficulties are being faced while rendering the images with different styles. This makes the image analysis a bit difficult process. The major limiting factor, in this case, is the lack of image representation that explicitly represents the semantic information, which complicates the representation of the image in a way that could further be used to separate the image's content part from the whole image. To resolve this issue, image representation derived from convolution neural networks (CNN) is used. In the current study, there is an advantage that CNN is optimized for object recognition, which enables it to make high-level image information explicit. This opens up the possibility for the application of a neural algorithm in artistic style which can separate and then recombine the image's content and its style. This enables us to combine information of random images with well-defined artworks and give it an artistic look that can further be used for different purposes. This deep analysis of images and its information is done by using CNN, and it portrays their capability of high-level image synthesis and image manipulation. The intermediate results in CNN can also be used for feature and content extraction from images.

Keywords Image analysis · Image representation · Convolutional neural network · Style transfer · Object recognition

---

S. Saurabh · B. Sahu (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: Sahu.barnali08@gmail.com

S. Saurabh
e-mail: Siddharth1997saurabh@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_13

1 Introduction

Image analysis involves a very tedious task as it involves defragmenting the image for a better and clear vision for the computer. The process is carried out by understanding the build of the image. One of the major steps of all is distinguishing the main content of the image from the background. This is started by reading the frequency in an image. We can classify an image's component into two categories, i.e., high-frequency and low-frequency components. High-frequency components are the ones where the intensity changes quite frequently, i.e., the level of brightness changes rapidly from one pixel to another. Low-frequency components may be one that is comparatively uniform in brightness or where brightness changes very slowly. Generally, the edges of objects in an image are accompanied by high-frequency components but the background is accompanied by low-frequency components. Edges in an image are those areas where the intensity change is very quick, and this often indicates object boundary. Next, more refinement of content and features of the image is done by high-pass filters. These high-pass filters convert an image to grayscale and identify areas of high-intensity pixel color change, coloring them white, and the rest low-intensity components as black [1]. As a result of this, the edges of the content in the image get highlighted. These high-pass filters are in the form of matrices often called convolutional kernels. A convolution kernel is a matrix of numbers that modifies an image and helps in edge detection [2].

Kernel convolution relies on centering a pixel and looking at its surrounding neighbors. If there are no surrounding pixels like an image corner or edge, then there are different ways to process the edges. The most common is to use padding, cropping, or extension [3, 4]. In extension, the border pixels of an image are copied and extended far enough to result in a filtered image of the same size as the original image. The nearest border pixels are conceptually extended as far as required, to provide values for the convolution. Corner pixels are extended in 90° wedges. Other edge pixels are extended in lines. Padding the image is padded with a border of 0's, black pixels. Cropping any pixel in the output image is skipped if it would require values from beyond the edge. This method might result in the output image being a bit smaller, with the edges cropped. All these steps are performed using a convolutional neural network (CNN) [5]. Convolutional neural networks (CNNs) are one of the main categories to perform the work of image recognition and its classifications. Object detection, face recognition, etc., are some of the areas where CNNs are mostly and generally used. In the current study, we have used CNN for style transfer of an input image.

The introductory section is followed by the following sections: Sect. 2 describes the dataset used, the architecture of CNN for image transfer, and the empirical set up for transformation. Section 3 discusses the experimental results. Section 4 draws a conclusion from the results.

2 Materials and Methods

2.1 Dataset Used

In this study, the initial style transfer image has been taken from the source [6, 7].

2.2 Model for Image Classification

As shown in Fig. 1, the CNN first needs to be trained by giving them different images for object identification or classification. But the image needs to be augmented in every possible way to make the training yield the best results. Image augmentation can be done in three possible ways which are scale invariance, rotation invariance, and translation invariance. Scale invariance means to scale the object's image in different sizes. Rotation invariance means to rotate the object's image at different angles and in different directions. Translation invariance means to position the object's image in different positions of the given space.

So CNN requires a huge database of object images to perform classification [8]. To overcome this difficulty, a pre-trained network called VGG-19 has been used in the current study. VGG was a work of Oxford University by the group named Visual Geometry Group in 2014 [9]. VGG-19 is a convolutional neural network which has been trained on more than one million images from the ImageNet Web site database [10]. The network is nineteen layers deep and besides that can classify images into thousand plus object categories, for example—keyboard, mouse, pencil, different animals, etc. As a result of training on such a huge amount of data, this network has learned very rich feature representations for a wide range of images. This network has an image input size of 224-by-224. VGG-19 has a very unique architecture.

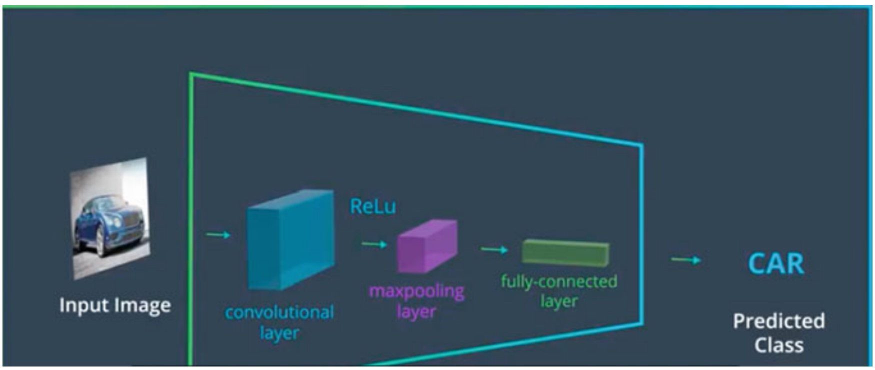


Fig. 1 Schematic representation of image classification using CNN

2.3 Convolutional Neural Network Training Process for Style Transfer

2.3.1 Layer Activation of CNN by Patterns

Very simple shapes and patterns are that causes the neurons in the first layer to activate. Like the lines shown below are patterns that activate the first layer in CNN as depicted in Fig. 2.

More complex patterns such as circles and stripes cause the neurons in the second layer to activate. The patterns shown below are some examples of patterns for second layer activation of CNN (Fig. 3).

Similarly, layer 3 detects patterns more complex than circles and stripes shown in Fig. 4.

More complex patterns such as circles and stripes cause the neurons in the second layer to activate. The patterns shown below are some examples of patterns for second layer activation of CNNs, the layer activation proceeds, and more and more complex patterns are detected layer by layer. Such as when the process reaches the fifth layer of CNN, these activated layers help us a lot in separating content from the image. The first layer of each of five convolutional stacks is responsible for content extraction.

Fig. 2 Sample pattern of image representation in the first layer of CNN

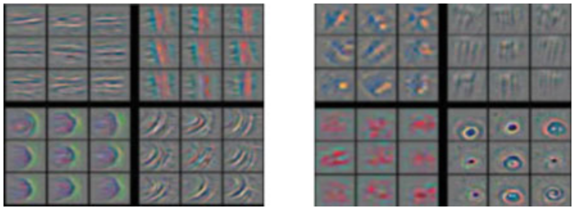


Fig. 3 Sample pattern of image representation in the second layer of CNN

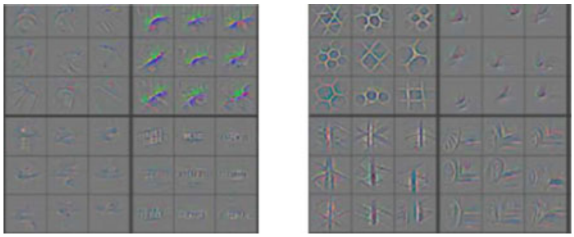


Fig. 4 Sample image form in the three layer of CNN

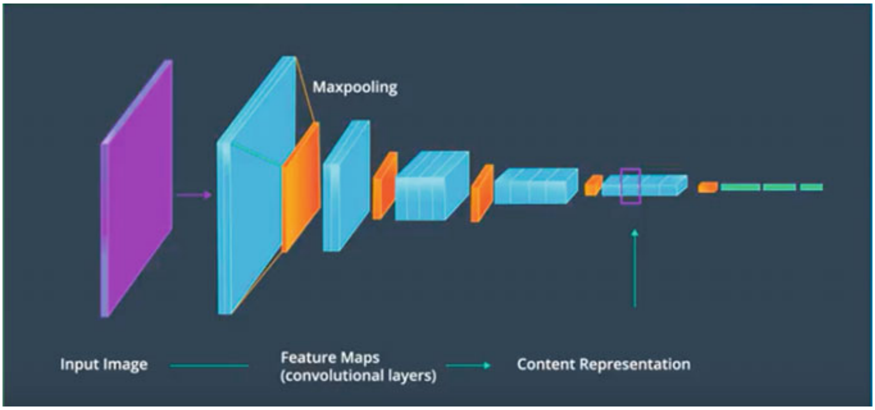


Fig. 5 Schematic representation of content representation in CNN

2.3.2 Content Representation

The content representation is taken as the output of the second layer of the fourth convolutional stack, i.e., layer conv4_2 shown in Fig. 5.

The new content representation of our formed target image after each cycle is matched with the content representation of our content image. Our approach should be to keep both the content representation of target image very similar to that of content representation of content image, i.e., content representation (content image and target image).

For the content representation of both target and content image to be the same, we need to minimize any loss in content. This loss in the transfer is called content loss.

2.3.3 Content Loss Calculation

Let the content representation for the content image be denoted by C.c and T.c be the content representation for the target image and L.content be the content loss. Then content loss is calculated by Eq. (1).

$$L.\text{content} = \frac{1}{2}\sum\left((T.\text{c} - C.\text{c})^2\right) \tag{1}$$

We aim to minimize the content loss, i.e., content should be as less as possible. For the content loss to be low, we should take care of the style that should not over fit our target image. In the case of overfitting of style over content, there will be a dropout and backpropagation.

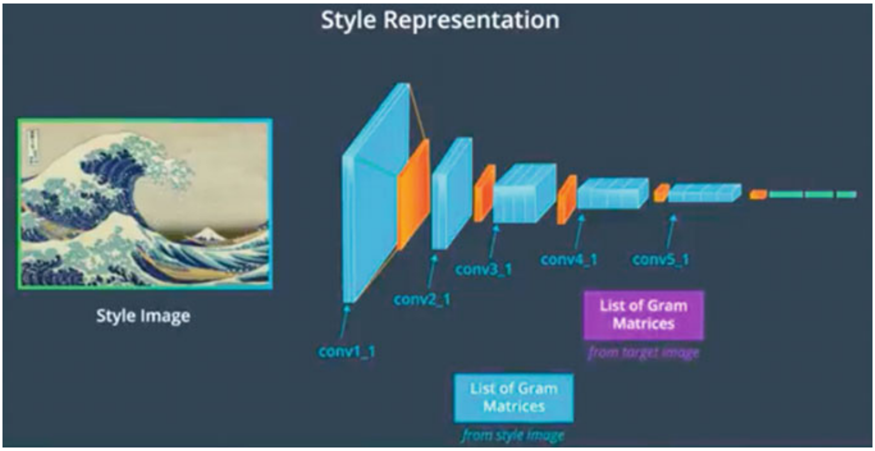


Fig. 6 Style representation of image in CNN

2.3.4 Style Representation

The VGG19 network is not used in the traditional sense, i.e., training is not being used to produce a specific output, rather a feature extractor and backpropagation are used to minimize a defined loss function between our target and style image. The style representation of an image relies on looking at the correlation between the features in individual layers of VGG19 network, i.e., we are looking at how similar the features in a single layer are. The style representation is calculated as an image passes through in the first convolutional layer in all five stacks shown in Fig. 6. By including the correlation between multiple layers of different sizes, it obtains a multiscale style representation of the input style image. The correlation between each layer is given by a Gram Matrix.

2.3.5 Max Pooling Using Gram Matrix for Texture and Colors

Gram Matrix, also known as style matrix, is a mathematical way of shared in prominent styles. It contains the non-localized information about the layer, i.e., the information will still be there if the image was shuffled around in space. Let us take a 4×4 image and convolve it with eight different image filters to create a convolutional layer. This layer will be 4×4 in height and width and 8 in-depth. So, we can say that the style representation for this layer has eight feature maps that we want to find the relationships between. Next, vectorizing this layer, it converts a 3D convolutional layer into a 2D matrix of values. Then, the 2D matrix is multiplied with its transpose matrix, and a value is obtained as shown in Fig. 7.

Finally, we are left with an 8×8 gram matrix whose values indicate the similarity between the layers.

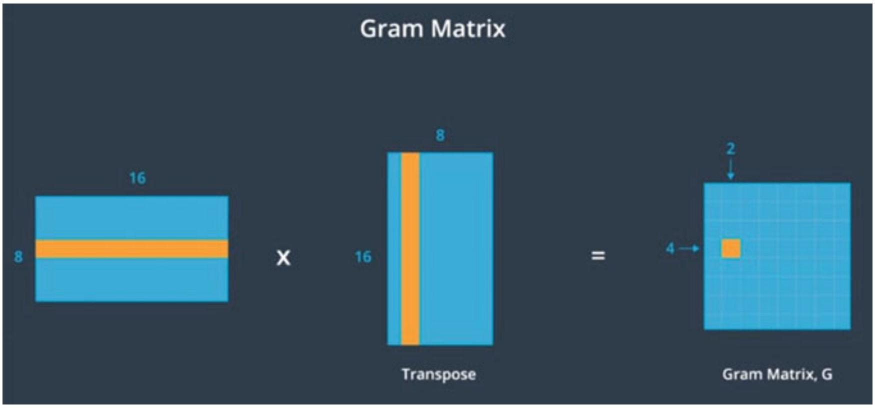


Fig. 7 Sample gamma matrix presentation

2.3.6 Style Loss

Let the style representation for the style image be denoted by S.s and T.s be the style representation for the target image and L.style be the style loss. 'a' is a value that accounts for the number of values in each layer. 'w' is style weights that we specify. Then, style loss is calculated by Eq. (2).

$$L.\text{style} = a \sum w_i (T.\text{s}_i - S.\text{s}_i)^2 \tag{2}$$

The aim is to keep the style loss, i.e., L.style as less as possible.

2.3.7 Total Loss

In total loss, we multiply weights, the content weight and, the style weight to L.content and L.style, respectively. Also is much larger than to give more weight to style for artistic style transfer.

$$\text{Total loss} = \alpha L.\text{content} + \beta L.\text{style} \tag{3}$$

The goal of total loss calculation is to minimize loss by backpropagation to iteratively change our target image to match our desired content and style. If there is any high trade-off between content and style, then we use the technique of dropout and backpropagation to prevent overfitting in any case. Also, the extra weights in the equation ensure that the trade-off between style and content is fair.

3 Experimental Results

The input content and style images are shown in Fig. 8.

3.1 Experimental Steps

1. Input content image and style image.
2. The output canvas in first iteration of model is set as the content image.
3. The VGG19 network recognizes the content and aims at minimizing the loss of content in further iterations.
4. VGG19 then keeps extracting the style from the style image and keeps imposing it over the style content of the content image.
5. The style is calculated using style matrix.
6. Balancing of style and content representation in the final image is done by allotting weights to both style and content representation of image.
7. With each iteration, the goal is to minimize the total loss and avoid overfitting in the process. If overfitting occurs, then the process needs to be dropped.

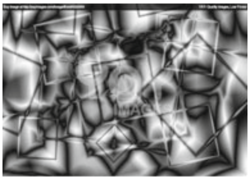
Input Content Image	Input Style Image
	

Fig. 8 Input content and style image

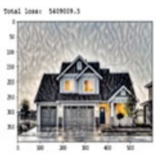
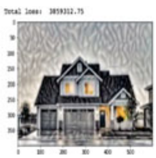
Epoch 400	Epoch 800	Epoch 1200	Epoch 1600	Epoch 20000
				

Fig. 9 Output images for 400, 800, 1200, 1600 and 20,000 epochs

We can observe at epoch 400 in Fig. 9, the content loss is 12.165156364440918 and the style loss is 17.4532260894775. But, as we gradually increase the number of epochs, the content loss increases and the style loss decreases (Table 1).

This is because the style from style image starts setting in the target image causing the content loss to increase in the style image. So, in the last epoch performed, i.e., epoch 20,000, the content loss increases to 15.28524398803711 and the style loss decreases to 2.8834359645843506. The total loss in the target image formed decreases irrespective of the trade-off between style and content in an image. The different approaches of performing style transfer are to change the deterministic optimizer from gradient decent to Ad delta. Other optimizers like Adam and Broyden–Fletcher–Goldfarb–Shanno (BFGS) is faster as compared to gradient descent. But, gradient descent being a much simpler concept of AI than the other optimizers, it is more preferred for small projects.

4 Conclusion and Findings

From the experimental analysis, it has been observed that the style on the content image from the style image sets and the total loss decreases with gradually increasing number of epochs. The key findings of this paper are that the content and style of an image can be separated and recombined to generate a new picture with either different content or different styles. The other findings include, if the content is to be maintained as it is during style transfer, then we need to compromise with the style; and if style is to be imposed, then we need to compromise with the content in the image. This trade-off can be minimized by adding weights to the content loss and weight loss. As we have inferred that the content of an image can be separated from its style and also can be recombined, instead of recombining content with style, we can combine content of one image with content of another. But once combined, separation will not be possible using this same style transfer concept.

Table 1 Loss for different epochs

Epochs	400	800	1200	1600	20,000
Total loss	17,453,238.0	8,453,548.0	5,406,244.5	3,854,168.75	2,883,451.25
Content loss	12.165156364440918	13.775644302368164	14.523797988891602	14.9713773727417	15.28524398803711
Style loss	17.45322608947754	8.453534126281738	5.4062299728393555	3.854153633117676	2.8834359645843506

References

1. Berning, M., Boergens, K.M., Helmstaedter, M.: SegEM: efficient image analysis for high-resolution connectomics. Neuron **87**(6), 1193–1206 (2015)
2. Cimpoi, M., Maji, S., Vedaldi, A.: Deep filter banks for texture recognition and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3828–3836 (2015)
3. Güçlü, U., van Gerven, M.A.J.: Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. J. Neurosci. **35**(27), 10005–10014 (2015)
4. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European Conference on Computer Vision, pp. 818–833. Springer, Cham (2014)
5. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2414–2423 (2016)
6. https://www.pexels.com/search/house(contentimage)
7. http://hdblackwallpaper.com/wallpaper/2015/08/black-and-white-abstract-art-3-desktop-wallpaper.jpg(styleimage)
8. Liu, L., Xi, Z., Ji, R., Ma, W.: Advanced deep learning techniques for image style transfer: a survey. Sig. Process. Image Commun. **78**, 465–470 (2019)
9. Ciresan, D.C., Meier, U., Masci, J., Gambardella, L.M., Schmidhuber, J.: Flexible, high performance convolutional neural networks for image classification. In: IJCAI, pp. 1237–1242 (2011)
10. Liu, S., Song, Z., Zhang, X., Zhu, T.: Progressive complex illumination image appearance transfer based on CNN. J. Vis. Commun. Image Represent. **64**, 102636 (2019)

Impact of Internet of Things (IoT) on 5G

Mitrabinda Khuntia, Debabrata Singh, and Sipra Sahoo

Abstract Currently, a disconnected system makes a major challenge for IoT technologies. The ability of 5G to transmit data more quickly and support further links can benefit at once address the present challenge and simplify the connected device control. In contrast, using 4G/LTE networks, 5G will be able to process data quickly, which was a challenge for IoT solutions. The consequence has been long delays from sending data to receiving it. By using the 5G network, more users could continuously send more information without fear of overcrowding the network, leading to delays in the past. The 5G connectivity would allow everyone to realize the IoT technology's strength. Now, IoT potential is vast, but the potential connectivity will come to fruition with 5G technology. Using sensors, "Smart" apps can easily transmit data even from thousands of miles away. In this paper, we discuss the impact and importance of 5G on IoT with its applications. As IoT is more established and essential due to the rapid growth of 5G, we discuss the establishment and necessity of IoT over 5G. Lastly, we focus on the aspect of IoT on updated 5G technology.

Keywords 4G/LTE · 5G · Internet of Things · Security · Artificial intelligence

M. Khuntia (✉) · S. Sahoo
Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: mitrabindakhuntia@soa.ac.in

S. Sahoo
e-mail: siprasahoo@soa.ac.in

D. Singh
Department of Computer Science and Information Technology, Institute of Technical Education and Research, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: debabratasingh@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_14

1 Introduction

In the last few years, the Internet of Things (IoT) has changed the pervasive measure including a large collection of built-up applications covering different sensor types. A large number of activities are expected to grow in the IoT-based product lines in the coming years of planning of up to billions of devices with an average of 6–7 devices per person by 2020 [1]. With most of the above device and protocol concerns resolved over the past decade, the cyber-physical and device-to-device (D2D) interaction convergence of sensors and sensor-based systems are now growing. A lot of discussion about 5G of the fifth generation of mobile communication networks has been taking place recently. As each new generation arrives, they are accompanied by more capacity and faster connections. Projected benefits include:

- Enhanced coverage: over the 4G and LTE towers, 5G cell towers will have improved capacity, which ensures that more phones will connect [2] simultaneously.
- Reduced latency: 5G eliminates log time and delays in sending and receiving information significantly.
- Faster connection: it is estimated that 5G speeds are about 10× faster than 4G connections are now feasible.

It is promised that the upcoming 5G wireless network technology will be responsive, fast and power-efficient [3]. While faster downloads and insufficient latency offered through 5G will attract mobile phone users, it is just not enough to explain the huge cost Bloomberg estimates at 200 billion a year.

The Internet of Things is still one industry that can see all of this differently as depicted in Fig. 1. According to projections by Bain and the firm, the B2B IoT market will reach 300 billion by 2020. And total IoT investment in 2022, based

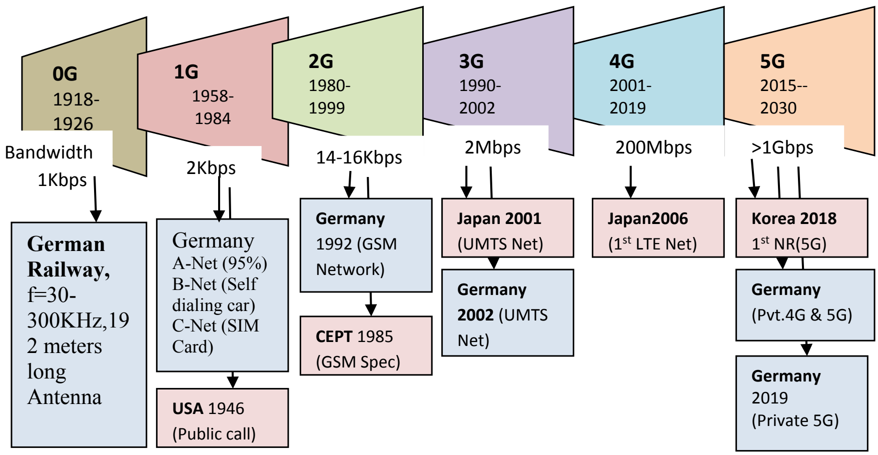


Fig. 1 Generation of private networks in Germany (from 0G to 5G in Germany)

on IDC results, is expected to reach 1.2 trillion [4]. The researchers, engineers and scientists face different challenges on IoT-based system development with 5G wireless communications. Mostly the advancement of cloud computing and its extension version fog computing, with the proliferation of smart IoT devices can be extended further. As IoT technologies, i.e. machine to machine communication, intelligent data analyses are expected to impact on the 5G. This research challenge helps us to design a new system with the presence the smart IoT devices and its applications.

The 5G network will mainly contribute to the creation of the Internet of Things as an integral part of the world by laying the foundation to unlock its full potential. Recent mobile technology has no equal accommodation for 20.4 billion connected devices and is setting up the exchange of data without small lags [5]. IoT's increasingly emerging technologies revealed an impressive step in five key areas in the technology interruption:

- Sensing—Endpoints of IoT,
- Communication—IoT communication,
- Secure—Secure IoT,
- Comprehension—IoT information and analytics,
- Acting—Intelligence Artificial IoT (AI).

5G promises a massive, friendly IoT ecosystem, with huge improvements over the 4G's current capacity. It is capable of covering up to 100 times more connected devices per unit area compared to the 4G LTE. IoT's idea is to have heterogeneous connected devices that collect the information in real-time over a limited period of time. After all, the continuous transmission of data contributes to exhaustion in the battery life of both the network and the phones. The new cellular network will see a 90% reduction in the network energy routine for up to 10 years of battery life for low-power IoT phones. Businesses and consumers of apps are enthusiastic about the prospects that will grow in 2019 and 2020 [6].

The rest of the paper is organized as follows: Sect. 2 presents the IoT challenges, which most likely offers the system requirements. In Sect. 3, we discuss 5G challenges. Section 4 presents 5G characteristics along with their applications. Section 5 describes the impact of 5G on IoT. Finally, in Sect. 6 we conclude the advantage of 5G over the use cases of IoT.

2 IoT Challenges

IoT offers everything systems with the requirement of software and a large number of items must be accessible at low cost. Therefore, IoT's problems and the core criteria are as follows:

1. **Energy efficiency**: Three important stages are harvesting, conversation and consumption of IoT ecosystem require energy. Novel energy-efficient solutions need to be explored [7]. Harvesting technologies and small batteries contribute

towards saving energy. But the problem occurs when the application takes place in a remote area where reaching is effort-consuming task and recharging or replacing batteries is another impossible task. When direct communication between IoT devices is established energy consumption may be reduced. Local connectivity is achieved by using short-range standardization wireless technologies and wireless gateways are adopted for providing remote connectivity to the Internet. The modern LTE-A cellular network is another contributor to energy-efficient networking solutions.

2. **Scalability**: Enormous statistics of smart devices are responsible to be part of the upcoming IoT world leading to the drawback of current network infrastructure. There are many drawbacks to be addressed in 5G based IoT systems though some recent works are carried out by 3GPP for supporting MTC in LTE-A. We expect percentage-based IoT architecture to include eliminating surplus in link access by maintaining efficient allocation of resource radio and efficient handling of small-size data communication on the assumption of high device performance.
3. **Resilience**: The dynamic essence of the wireless IoT ecosystem ensures system operation under severe conditions including deficiencies in connectivity to the network infrastructure. Unlike the capillary network broadcasting successes, due to crowded incidents, network node failure situations, not so good wireless networking conditions, and many catastrophic circumstances, an unforeseen lack of infrastructure base is likely to occur. In addition to the lack of network infrastructure access, the inherent dynamic nature of wireless IoT environments often needs device stability assurances under harsh conditions.
4. **Interoperability**: Highly heterogeneous artifacts populate the IoT, each having specific functions that are accessible through its own language and network one of the main criteria, is to handle this inherent diversity, which provides efficient solutions for the seamless integration of devices, technologies and services [8]. From the communication point of view, IoT versatility will take the range of radio technologies involved in the support of low-power devices. In 5G network, all evolving trends are supported by the connectivity of IoT applications. For those next-generation cellular networks needs an effective mechanism to handle heterogeneous data capabilities, manages different radio technologies and integrated mobility management, etc.
5. **Team communications**: In IoT universal environments, data provided by a single object may not be reliable or useful enough to support specific applications and the required quality of the data. At the same time, autonomous IoT systems can have advantages in activating simultaneous actions on multiple actions such as street light lamps in a smart city. The relevance of IoT group communication involves standardizing a resource-constrained device application protocol based on IPv6 [3]. Multicast and unicast-oriented solutions can provide team communications. The former situation is the most difficult because the network has to facilitate the simultaneous transmission of packets to a number of recipients. This enables network traffic to be reduced and the efficient use of resources to be improved [8].

6. **Cloud-based IoT network environment**: Supporting a dynamic execution system for complex IoT applications is another major challenge. On-demand processing and storage tools are deployed globally with supported by data centers. IoT device virtualization, dynamic processing sensor events, image transcoding, face reorganization, storing huge amount of data and analyzing the big data make more challenges in IoT platforms. Vehicular networks [9], Fog computing [6] was introduced for IoT, address these issues through cloud services. These cloud computing solutions face delays in communication with remote data centers due to traffic congestion.
7. **Multimedia IoT support**: Multimedia smart devices also need to be fully incorporated to support multimedia services in order to implement a robust IoT platform. Different use cases involve telemedicine-based ambient assisted living and patient care, integrated smart home monitoring systems, advanced interactive smart city monitoring involving real-time processing of sensor data. In addition, the so-called "Multimedia Things Internet" [7] incorporates functionality and network specifications that vary from those of the traditional resource-limited IoT environment. Multimedia things foresee higher computational capabilities to manage multimedia flows and, above all, communication is focused on bandwidth, jitter, and loss rate to ensure acceptable multimedia content delivery. Low-power radio systems are not well suited to serve these traffic types, although cellular networks offer better performance for multimedia flows. Nevertheless, taking into account the additional traffic created by multimedia products, 5G must include new, efficient strategies to satisfy both system and human requirements, e.g. by utilizing edge data caching and delivery of proximity information.

3 5G Challenges

In 5G development, the early challenges include:

- Using different frequency bands: In 5G networks a combination of low, medium and high channels. All the mobile operators are expected to be implementing 5G services through a smaller spectrum band.
- 4G to 5G Gradual transition: In industry, services must be 4G to 5G, due to the critical continuous requirements of on-demand delivery of hardware, software and services.
- Preparing the core network: Implementing the 5G setups, requires substantial improvements in cloud computing networks, in the area of virtualization and MIMO.
- Data interoperability: It is very crucial to achieve interoperability between user elements (UE) and commercially developed 5G networks to validate key technology.
- 5G business models Establishment: In industries, 5G business models must be low-cost and high-performance implementations as depicted in Fig. 2. Developing

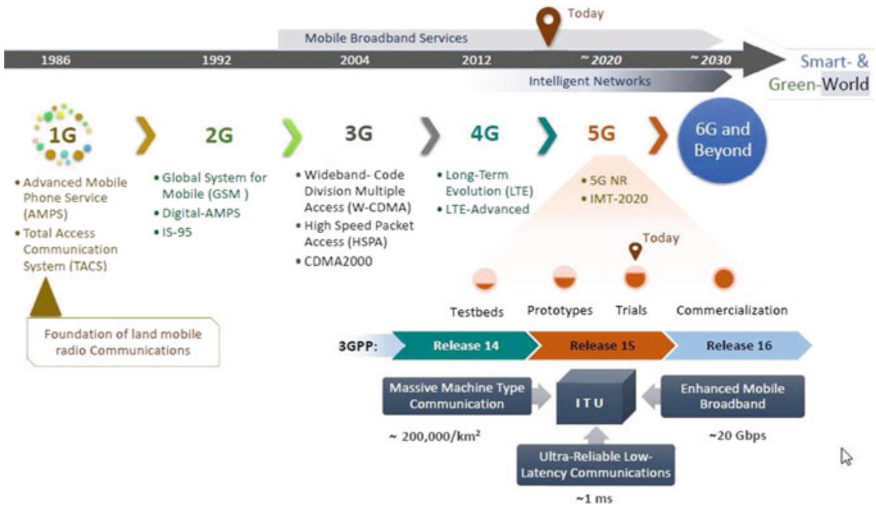


Fig. 2 Evolution of mobile technology [8]

a network infrastructure and application ecosystem to help a sustainable business model for 5G services.

4 5G Characteristics and Their Values/Applications

There are some characteristics of 5G along with their values/applications are summarized in Table 1.

4.1 5G on IoT: A Single Network for Millions of Applications

With commercial 5G deployments beginning across the world, there will be increasing interest surrounding the benefits that 5G can provide to the growing Internet of Things (IoT) movement. A major roadblock to realizing the potential and promise of the IoT is that multiple, specialized networks are being utilized for different IoT use cases, from applications that utilize low-data transfer rates to high-end mission-critical applications that require instantaneous data transfers. 5G, however, offers a solution that allows for a single network to support myriad IoT use cases and build economies of scale.

Table 1 Characteristics and their application

Characteristics	Application
Network characteristics	Cloud computing, software engineering, virtualization, slicing
Maximum data rate	20 Gb/s
Maximum experienced data rate	0.1 Gb/s
Efficiency rate	3 times of 4G
Network efficiency	10–100 times of 4G
Traffic capacity	10 Mb m^2/s
Density of the connectivity	10^6 devices/km^2
Latency	1 ms
Mobility	500 km/h
Technology	Cloud/fog/edge computing, massive MIMO, flexible frame structure, network slicing
Usage scenario	eMBB, URLLC and mMTC

4.2 The Importance of 5G on the IoT

In short, IoT growth is exploding, and using multiple specialized networks to handle various IoT applications is costly and difficult to scale. By 2020–2030 in a period of just ten years, IoT devices will expand by 40–140 billion and the IoT upgrade from 4G to 5G is strong: today's 4G networks, for instance, can accommodate up to 6000 NB-IoT devices on one cell. With 5G, on the other hand, a single cell can handle up to one million devices. IoT applications that require minimal data transfer rates can result in massive volumes of data transmitting over networks, and this requires a great deal of connection management for each network. On 5G, however, this is not the case: the single network approach of 5G is already optimized to handle massive data transfers across a broad range IoT application [10].

4.3 Application of 5G Over IoT

5G and IoT together would also help to put each product on the shelves to the Internet. Consumer products do not need to be continuously connected to the Internet as hardware devices, but they can send and receive data about themselves as connected smart products based on event-based experiences with clients and other entities through scanning, RFID readers, NFC tags and more [11]. The current wireless infrastructure is not up to the task of dealing with so many network devices, but 5G will make it possible. Smart Packaging and Digital Labels can transform the way retailers manage inventory and logistics and provide a hotbed of imagination to use them as a way to interact with consumers in a creative manner. 4G does not manage data load from the ever-increasing number of online sensors and connected devices, limiting what IoT can actually do. The 5G is the ideal enabler for the Internet of Things with its high data speed, low latency, increased mobility, low energy consumption, cost efficiency and the ability to handle much larger devices. 5G can play a major role not only in transforming the way we communicate but also in changing industry and society. There are a number of companies in which 5G as well as IoT can cause interruption together, that are:

1. **Self-driving cars**: Sensors generate large quantities of data on self-driving cars, temperature measurement, traffic conditions, weather, GPS location, etc. The collection and assimilation of each quantity of data require a great deal of energy. These cars also heavily relay on real-time information transmission in order to provide optimum services. Nevertheless, with high-speed communication and low latency, this intelligent care will be able to collect all kinds of data on an ongoing basis, including time-critical data on which algorithms will work independently to keep track of the working condition of the car and improve future designs.
2. **Healthcare**: As all types of medical devices are powered by IoT, changes in their services will also be seen in the medical field. Notwithstanding proper healthcare infrastructure, the IoT link will greatly benefit rural areas and other similar remote locations. With such low latency, it becomes an option to provide world-class health care services such as remote surgery [12].
3. **Logistics**: 5G networking will improve end-to-end logistics operations with advanced IoT monitoring sensors. High speeds and low latency will not only allow data to be obtained in real-time, but also enable energy efficiency to generate more diverse information at all points within a supply chain for a very long time. A buyer would have access to detailed information such as where the fish she had just bought was caught, the temperature at which it was treated during processing, and when delivered to the seller.
4. **Smart cities**: 5G will allow broader applications from water and waste management to smart city projects, traffic control to enhanced facilities for health care. Smart cities will benefit from the benefits of the new generation network as more and more devices reach urban infrastructure. Not only will 5G be able to handle the massive data load, it will also make it a reality to incorporate multiple smart systems that continuously interact with each other, bringing a truly connected city's dream closer.
5. **Retail**: As they attempt to shape customer engagement and experience through mobile phones, retail IoT will see a positive impact from 5G's arrival. Improved connectivity and a larger number of network-connected devices would allow new and innovative ways of engaging consumers to engage faster with shoppers through better digital signage. With increased reality and virtual reality, it will become more popular. Retailers will be able to enhance the shopping experience by implementing omnichannel sales activities more efficiently.
6. **Automotive**: It is one of the main uses of 5G connecting cars to Augmented Reality (AR) and Virtual Reality (VR) [10]. Enhanced vehicle communication

services will include direct vehicle-to-pedestrian and vehicle-to-infrastructure communication, as well as autonomous driving communication that is network-friendly. Supported use cases would concentrate on vehicle comfort and safety, including real-time communication of purpose, route planning, organized driving and community updates.
7. **Industrial**: By incorporating 5G security into the core network architecture, we would also provide an extremely secure network for industrial IoT.

5 The Impact of 5G on IoT

Today, disconnected networks are a major challenge for IoT technologies. The capacity of 5G to transmit data more rapidly and allow more connections will help at once address this issue as well as simplify the management of connected devices. In contrast, 5G will be able to process data quickly using 4G/LTE networks, which has been a challenge for IoT solutions. The result was a long delay from the time the data was sent to the time it was received. The 5G connectivity would allow everyone to understand the IoT technology's strength. As of now, IoT's potential is enormous, but the real networking must come to fruition with 5G technology. Using sensors, "Smart" apps can easily transmit data even from thousands of miles away. The implications on an individual and municipal scale are endless. The 'smart' city has become a reality that will reap the rewards for both local businesses and residents [11].

5G will make it possible for companies investing in IoT technology or creating IoT-based platforms to have many of the desired specifications. Better connectivity, lower latency and faster connection mean more people can transmit more data at the same time. As a result, IoT solutions will grow companies constantly without thinking about disconnected networks that have plagued IoT developments so far. 5G facilitates the development of IoT applications to help all.

5.1 Establishing a 5G—IoT Ecosystem

IoT will have more chances to expand capabilities; services as well as reliability as more development resources like 5G enter the market. According to Statista, "The Internet of Things Devices installed base is expected to grow to nearly 31 billion worldwide by 2020" [12], Factors needed to build a 5G-IoT ecosystem are as follows:

1. Automatic power supply: Batteries and wires may be a viable IoT power solution now, but it will be nearly impossible to keep up as IoT's sheer volume expands globally. A failed or drained battery can cost a company's revenue and increase protection and liability concerns in an IoT sensor, M2M or factory automation. Having wireless power without pads and over the range is important.

2. Innovators, integrators and implementers: The implementation of 5G is clearly expensive. Companies need a short and long term growth strategy along with people to help them fully realize the benefits, safe and secure [13].
3. A electronic recycling program: Normally we will keep our home appliances and work equipment for dozens of years. With IoT enabled by 5G and a continuous and automatic IoT sensor wireless power supply, existing "non-smart" devices will soon become obsolete and need recycling or upcycling. Industries and societies that prepare differently for this will become an opportunity to present a potential threat to the environment.

5.2 Future of IoT

- By 2025, it is estimated that more than 21 billion IoT devices are in the market.
- Cybercriminals will continue their role with IoT devices through DoS and DDoS attacks.
- More cities will seem smarter [14].
- Artificial intelligence with networking will become a bigger challenge in the upcoming scenario.
- Routers will continue to become more secure and smarter.
- Instead of 4G, 5G networks will continue to fuel IoT growth.
- Vehicular networks, i.e. Cars, trains, buses will get even smarter [15].
- 5G's arrival will help in smart home automation and also act to open the door with privacy and security concerns.
- IoT-based DDoS attacks will create a major problem in the networks [16].
- Security and privacy concerns will drive legislation and regulatory activity.

5.3 Aspects of IoT on Updated 5G Technologies

There are some of the different aspects of IoT to be enhanced by 5G technology, which are as follows:

Data The most obvious impact 5G technology implementation will have on the IoT is its ability to share far larger data volumes at faster speeds than 4G. Using more advanced communication techniques, such as MIMO, to improve 5G networks ensures that more information can be sent and retrieved in a relatively short time frame. Numerous transmitters and receivers distributed over large areas work much better than the distribution of single antennas. Hence, coverage is usually more difficult to reach remote rural areas or large buildings inside that can certainly be enhanced.

Size In the previous years, the IoT's sheer size has improved significantly as those devices are linked and additional applications are being created. 5G networks' ability to transmit more data at faster speeds would empower more connected devices to connect and talk to each other to the network. High latency has been a continuous problem for companies with multiple connected devices, but after adapting the 5G network, companies have been able to add many more devices to their network without causing congestion and further latency problems.

Power Consumption Another issue that concerns organizations and individuals planning to implement multiple connected devices to a system. For example, recent developments in the narrowband IoT enable narrow bandwidth that is optimized for low-data rates IoT applications. This would empower the network with much lower power consumption and reduce the strain on data transmission as well. For many industrial and business operations, scalability is key and 5G technologies will hopefully make such conceivable results viable.

6 Conclusion

Over the past few years, there has been an increase in mobile broadband technology, 2G networks were designed for voice communication, 3G networks added voice and data and 4G offered a boost to Internet-based broadband experiences. 5G is about fusing networking computing capabilities with imagine a world in which connected devices do not have to take the computing load because the network they communicate over is capable of processing enough. 5G will also help to realize IoT's potential well beyond what is possible with today's technologies. Human and object interactions will increase to all-new levels. 5G will provide countless benefits on the road to realizing the potential of the IoT. The advantage of using a single 5G networks will prove more efficient, more cost-effective and will provide economies of scale across a wide variety of IoT use cases. 5G is believed to be more speed up to 10 Gbps, lower latency and higher extensive coverage and increase the data traffic protection.

References

1. Sinha, R.S., Wei, Y., Hwang, S.-H.: A survey on LPWA technology: LoRa and NB-IoT. ICT Express **3**(1), 14–21 (2017)
2. Ortiz, S.: 4G wireless begins to take shape. Computer **40**(11), 18–21 (2007)
3. Liu, J., Wan, J., Jia, D.Y., Zeng, B., Li, D., Hsu, C.-H., Chen, H.: High-efficiency urban-traffic management in context-aware computing and 5G communication. IEEE Commun. Mag. **55**(1), 34–40 (2017)
4. Liu, X., Effenberger, F.: Emerging optical access network technologies for 5G wireless. IEEE/OSA J. Opt. Commun. Netw. **8**(12), B70–B79 (2016)
5. Velev, D.G.: Internet of things: analysis and challenges. Econ. Altern. **2**, 99–109 (2011)
6. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, pp. 13–16. ACM (2012)

7. Singh, D., Pattanayak, B.K., Satpathy, P.R.: Ambient energy harvesting and management on the sensor nodes in a wireless sensor network. Int. J. Renew. Energy Res. (IJRER) **7**(4), 1869–1879 (2017)
8. Tudzarov, A., Janevski, T.: Functional architecture for 5G mobile networks. Int. J. Adv. Sci. Technol. **32**, 65–78 (2011)
9. Raya, M., Hubaux, J.-P.: Securing vehicular ad hoc networks. J. Comput. Secur. **15**(1), 39–68 (2007)
10. Billinghurst, M., Kato, H.: Collaborative augmented reality. Commun. ACM **45**(7), 64–70 (2002)
11. Alvi, S.A., et al.: Internet of multimedia things: vision and challenges. Ad Hoc Netw. **33**, 87–111 (2015)
12. Stergiou, C., Psannis, K.E.: Recent advances delivered by mobile cloud computing and internet of things for big data applications: a survey. Int. J. Netw. Manage. **27**(3), e1930 (2017)
13. Chen, M., Zhang, Y., Hu, L., Taleb, T., Sheng, Z.: Cloud-based wireless network: virtualized, reconfigurable, smart wireless network to enable 5G technologies. Mob. Netw. Appl. **20**(6), 704–712 (2015)
14. Mishra, K.N., Chakraborty, C.: A novel approach toward enhancing the quality of life in smart cities using clouds and IoT-based technologies. In: Digital Twin Technologies and Smart Cities, pp. 19–35. Springer, Cham (2020)
15. Chakraborty, C., Gupta, B., Ghosh, S.K.: A review on telemedicine-based WBAN framework for patient monitoring. Telemed. e-Health **19**(8), 619–626 (2013)
16. Rath, M., Swain, J., Pati, B., Pattanayak, B.K.: Network security: attacks and control in MANET. In: Handbook of Research on Network Forensics and Analysis Techniques, pp. 19–37. IGI Global (2018)

Prediction of Exchange Rate in a Cloud Computing Environment Using Machine Learning Tools

Trilok Nath Pandey, Tanu Priya, and Sanjay Kumar Jena

Abstract In this ever-changing world, cloud computing will definitely play a vital role in our daily life as everything is going to be mapped on it. In this paper, we have considered the cloud environment provided by Microsoft Azure to use machine learning tools for the prediction of exchange rate. We have also considered the exchange rates of forty-two different countries to design our data set for the experimental purpose. We have used various machine learning tools provided by Microsoft Azure and predict the future values of dollar in terms of Indian rupees (US/INR). The experiments are performed in the cloud computing environment, and it has been analyzed that neural network regression model is performing better than the other models considered in this paper. It has also been observed that the performance of the machine learning tools provided by Microsoft Azure are very much competitive with respect to the traditional machine learning approach.

Keywords Cloud computing · Neural network regression · Decision tree regression · Exchange rate · Random forest regression · Boosted tree regression

1 Introduction

Humans have been using the concept of "prediction" since ancient times. In past, this use was restricted to non-scientific fields. But in today's generation, its usage has expanded to almost all horizons of science and economy. One such application of prediction models is in the exchange rate predictions in the international business

T. N. Pandey (✉) · T. Priya · S. K. Jena
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: trilokpandey@soa.ac.in

T. Priya
e-mail: tanupriya604@gmail.com

S. K. Jena
e-mail: sanjayjena@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_15

market. Prediction can be termed as the forecast of an event or an occurrence that may happen in the future. It is purely an expectation about the future values of a variable taking into consideration the current conditions of an entity which has a link with the variable that has to be predicted. Prediction is important since it gives a short-term forecast of the near future which plays a vital role in minimizing losses or even maximize profit, many times [1, 2]. That is the reason why prediction has become an integral part of business and economic structure in present-day scenario. With passing years, prediction in exchange rates has become an old affair now [3]. There are traditional ways of exchange rate predictions that are still in use in the international business environment.

1. **Purchasing Power Parity (PPP)**
   The basis of PPP is the "Law of One Price." This law says that identical goods should have identical prices, irrespective of the region or the country where they are being sold. PPP also accounts for inflation from country to country.
2. **Relative Economic Strength Approach**
   The relative economic strength approach is a generalized assessment of a country's currency rates. It looks to the growth of economy in a given country. The pool of investors attracted by an economy is directly proportional to the strength of that economy. Also, high interest rates are a good sign for people investing in that economy.
3. **Econometric Models**
   In econometric models, an economic factor that would have an effect on currency is picked and then a model based on that factor is created. Thus, such models can be very complicated, at times.
   As we saw above, the traditional ways to predict exchange rates follow an approach of defining all the steps to reach the desired outcome, which makes the process of prediction quite complicated. On the other hand, using machine learning techniques in prediction process flips the approach completely [4]. With machine learning techniques, we define the outcome and the program learns the steps to reach it. It involves letting our algorithm learn the steps to reach the expected outcome [5].

2 Machine Learning Techniques Used for Prediction Modeling Data

In this paper, we have used various machine learning techniques to predict the exchange rate of US dollar. Some of the techniques have been discussed below.

2.1 Linear Regression

Linear regression tries to model a relationship between two variables by finding a linear equation based on the observed data set. One of the variables is an independent variable while the other one is a dependent variable. Dependent variable is the variable whose value is required for explaining our forecast and independent or explanatory variable explains other variables. Mathematically, it can be represented as

$$Y = MX + B \tag{1}$$

B Intercept
M Slope or gradient.

2.2 Regression Trees

Regression trees are used when dependent variable is continuous, and they use mean or average. In regression trees, the terminal nodes denote the mean or average value of the observation [6].

2.3 Decision Tree Regression

Decision tree is a tree structure that is used as a decision-making tool. The nodes of trees represent either conditions or results, and the edges or branches denote the result of the node. This regression technique observes the features of an entity and then trains a model in the structure of a tree to make predictions [6, 7]. Using this model, continuous values are predicted.

2.4 Boosted Trees Regression

Boosted trees model is a kind of additive model. This regression model combines decisions from a sequence of base models to make predictions, such that each base model is nothing but a simple decision tree [8]. It uses a particular model ensemble technique called gradient boosting using multiple models to obtain better predictive performance. Unlike linear regression models, the boosted trees regression model can capture even the nonlinear interactions between the features and the target [6, 9–11].

2.5 Random Forest Regression

It is the ensemble technique which is capable of performing regression, as well as, classification tasks using multiple numbers of decision trees and a machine learning technique called bagging. Bagging involves training each decision tree using a different subsample of the data. It improves the stability and accuracy. More trees in the forest, the more robust the prediction, thus higher accuracy. It handles the missing values and maintains accuracy for missing data when a large proportion of the data is missing. It handles larger data set with higher dimensionality [6, 9].

2.6 Neural Network Regression

This method involves the creation of a regression model using the neural network algorithm. Neural networks are widely known for their use in modeling of complex problems which includes regression problems as well [12, 13]. Many of the things we do every day involves recognizing patterns and using them to make decisions, so neural networks can help us out in various different ways. One of its major applications being exchange rate predictions. A basic artificial neural network consists of three layers, named as input layer, hidden layer and output layer [14–17]. Each of the three layers consists of single or more than one nodes, generally represented by small circles as shown in Fig. 1. The lines connecting the nodes indicate how the information flows from one node to the next.

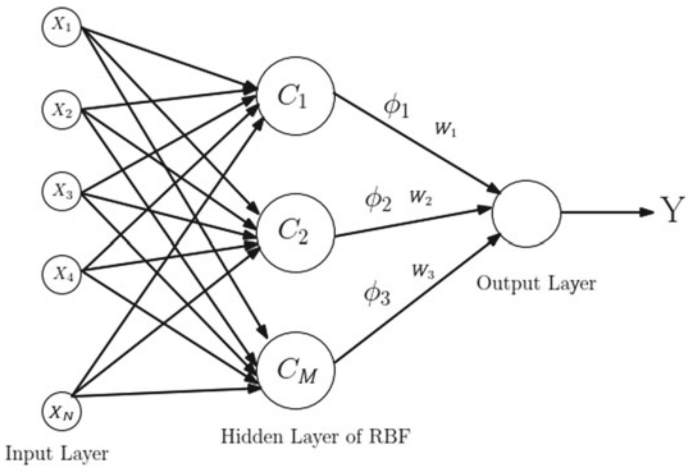


Fig. 1 General architecture of neural network

$$E = \sum_{j=1}^{p} \left(y_j - \hat{y}_j\right)^2 = \sum_{j=1}^{p} \left(y_j - \sum_{i=1}^{m} w_{ij}\phi_i(x)\right)^2,$$
$$\phi_i(x) = \exp[-\|x - c_i\|]^2 / \left(2\sigma^2\right). \quad (2)$$

2.7 Cloud Computing Tools

Cloud computing is basically a platform that provides access to a number of computing resources over the Internet. Computing resources may include serverless computing, virtual machines, storage and many more. The users actually connect to these data centers to collect their data or use it. Users may access any service based on their requirements. Cloud computing is used for a variety of applications, including the following services:

1. Machine learning and data analysis
2. Data storage and backup
3. Streaming media content
4. Creating and testing applications
5. Automating software delivery
6. Hosting blogs and applications.

These services are provided by various cloud service providers such as

1. Microsoft Azure
2. Amazon Web Services (AWS)
3. Google Cloud Platform
4. IBM Cloud Services.

These cloud organizations can deploy and access its resources such as storage, networking and compute power with the help of the Internet. From the above-mentioned description, we could easily conclude that various cloud service providers are best suited for varying applications based on the services provided by them respectively. Since exchange rate prediction experiments may require data sets from different distant sources, geographically, and these data sets are ought to be dynamic in nature, Microsoft Azure is preferred over other cloud service providers since it has datacenters in 42 regions around the world, the most for any cloud service provider at the moment. It is important for any cloud service provider to have many data centers around the world because this makes it possible to reach a wider audience.

3 Experimental Works

In this section, we have discussed the detail analysis of experimental work and data set analysis.

3.1 Data set Analysis

In this paper, we have collected exchange rates of forty-two different countries with respect to Indian rupees for prediction purpose. All the currency rates are collected from the following website: http://www.bankofcanada.ca/rates/exchange. The data are collected from February 27, 2006, to February 25, 2016. We have considered 70% of our data for training purpose, and the rest are used for testing purpose. We have also normalized the data set column wise within the range 0–1 using min–max normalization technique. From the Pearson and Durbin–Watson analysis of financial data, we have found that exchange rates of different countries are correlated with each other. Pearson coefficient and Durbin–Watson statistic of most of the G7 country's exchange rates with respect to exchange rate of major countries lies well within 0.7–0.99 and 0.181–0.732. From this analysis, we have concluded that we can use exchange rates of different countries as input to the network for prediction purpose. Thus, we can avoid the use of tradition indicators and the factors that affect the exchange rate.

3.2 Result Analysis

In this article, we have used the cloud computing environment provided by Microsoft Azure. Several machine learning tools provided by Azure such as boosted decision tree, decision forest tree, linear regression and neural network regression have been used to predict the future value of US dollar [18–20]. In this experiment, we have considered several performance metrics for future prediction of US dollar, and the results have been presented in Table 1. Graphically, the prediction results of different machine learning techniques for the testing data set have been depicted in Figs. 2, 3, 4 and 5. We have analyzed that the performance measure of neural network regression machine learning is better in comparison to other algorithms we have considered in this paper.

Table 1 Comparison of machine learning techniques provided by Azure

Machine learning techniques	Coefficient of determination	Mean absolute error	Relative absolute error	Relative squared error	Root mean squared error
Linear regression	0.999989	0.005494	0.000776	0.000031	0.00788
Decision forest tree	0.999957	0.024736	0.003492	0.000043	0.05306
Boosted decision tree	0.999759	0.089896	0.012692	0.000241	0.12624
Neural network regression	0.999999	0.000463	0.000058	0.000011	0.00073

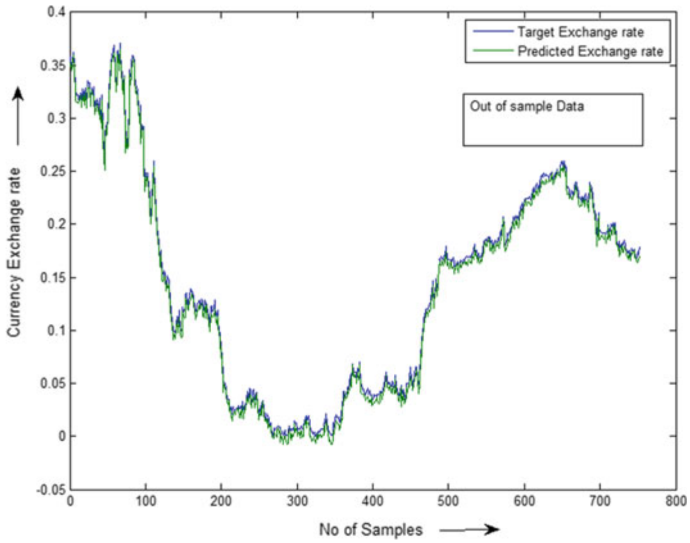


Fig. 2 Prediction of US/INR using linear regression (out of sample data)

4 Conclusion

Presently, the prediction of exchange rates in accurate and effective way is one of the major challenges being faced by the global economy, mainly, because of increasing large amounts of data and their dynamic nature. In this scenario of dynamic currency exchange rates, cloud computing becomes a crucial part of the whole process. An exchange rates prediction system based on cloud computing is adopted for collection for real-time exchange rates information and the prediction of future exchange rates in efficient computing time. In this cloud computing environment of Microsoft Azure,

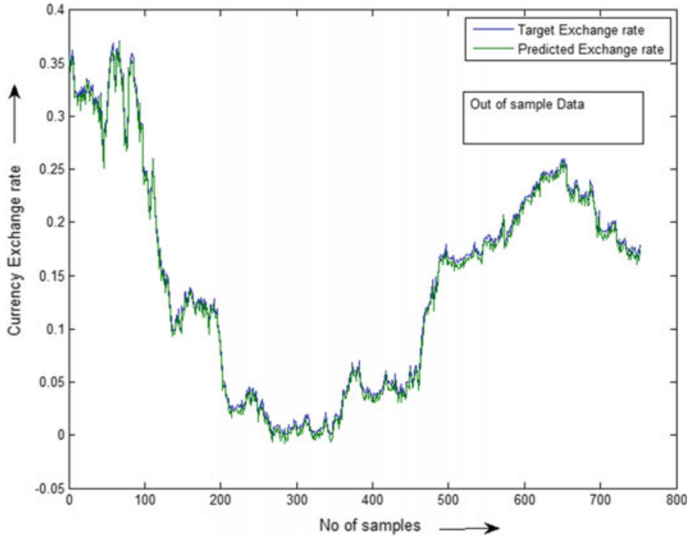


Fig. 3 Prediction of US/INR using decision forest tree regression (out of sample data)

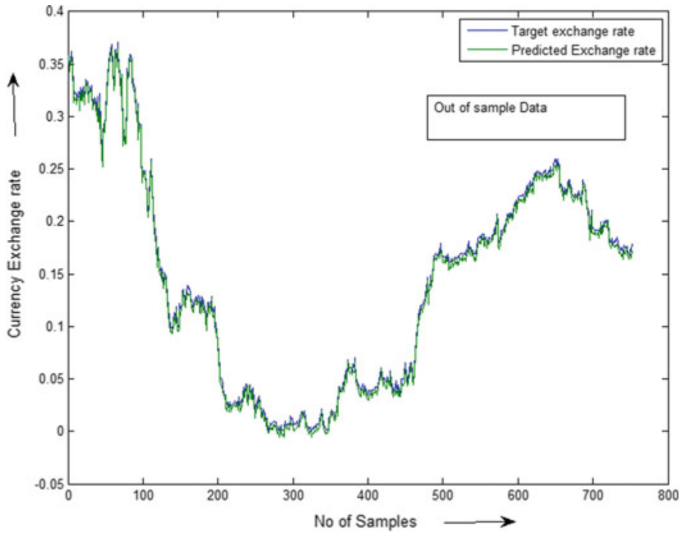


Fig. 4 Prediction of US/INR using boosted decision tree regression (out of sample data)

it has been observed that neural network regression model is outperforming the other models such as linear regression, decision forest tree regression and boosted decision tree regression. In future, more precise and accurate performance can be achieved by combining machine learning tools with latest artificial intelligence technologies.

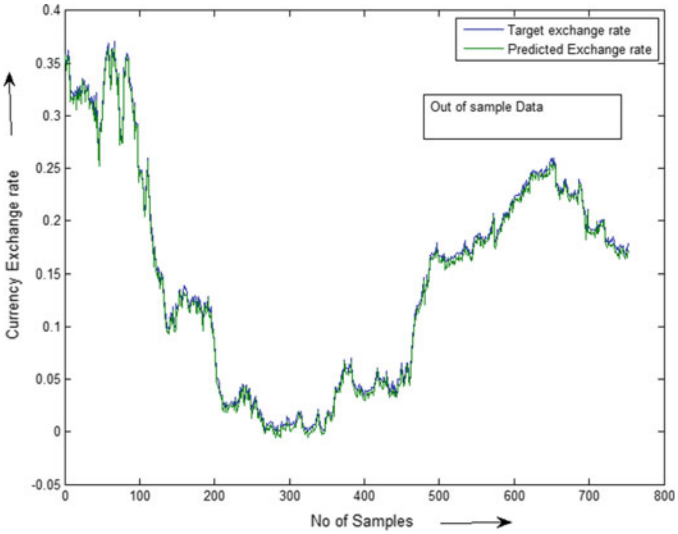


Fig. 5 Prediction of US/INR using neural network regression (out of sample data)

References

1. Dominguez, K.: Are foreign exchange forecasts rational? New evidence from survey data. Econ. Lett. **21**, 277–282 (1986)
2. Zhou, F., Peng, H., Qin, Y., Zeng, X., Xie, W., Wu, J.: RBF-ARX model-based {MPC} strategies with application to a water tank system. J. Process Control **34**, 97–116 (2015)
3. Abdelhamid, B.: Radial basis function nets for time series prediction. Int. J. Comput. Intell. Syst. **2**(2), 147–157 (2009)
4. Ak, O.F., Zio, E.: Two machine learning approaches for short-term wind speed time-series prediction. IEEE Trans. Neural Netw. Learn. Syst. **27**(8), 1734–1747 (2016)
5. Bayona, N.F., Fornberg, B., Barnett, G.A.: On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs. J. Comput. Phys. **332**, 257–273 (2017)
6. Pandey, T.N., Jagadev, A.K., Dehuri, S., Cho, S.-B.: A review and empirical analysis of neural networks based exchange rate prediction. Intell. Decis. Technol. **12**(4), 423–439 (2019)
7. Pandey, T.N., Jagadev, A.K., Dehuri, S., Cho, S.-B.: A novel committee machine and reviews of neural network and statistical models for currency exchange rate prediction: an experimental analysis. J. King Saud Univ.-Comput. Inf. Sci. (2018)
8. Trucos, C., Hotta, L.K.: Bootstrap prediction in univariate volatility models with leverage effect. Math. Comput. Simul. **120**, 91–103 (2016)
9. Yetis, Y., Kaplan, H., Jamshidi, M.: Stock market prediction by using artificial neural network. In: World Automation Congress (WAC), pp. 718–722 (2014)
10. Zhong, H., Miao, C., Shen, Z., Feng, Y.: Comparing the learning effectiveness of BP, ELM, I-ELM and SVM for corporate credit rating. Neurocomputing **128**, 285–295 (2014)
11. Pandey, T.N., Jagadev, A.K., Mohapatra, S.K., Dehuri, S.: Credit risk analysis using machine learning classifiers. In: 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), pp. 1850–1854 (2017)
12. Mulder, D., Bethard, S., Moens, M.: A survey on the application of recurrent neural networks to statistical language modeling. Comput. Speech Lang. **30**, 61–98 (2015)
13. Guresen, G.K., Daim, T.U.: Use of artificial neural network models in stock market index prediction. Expert Syst. Appl. **38**, 10389–10397 (2011)

14. Kruppa, J., Schwrz, A., Arminger, G., Zeigler, A.: Consumer credit risk: individual probability estimate using machine learning. Expert Syst. Appl. **40**, 5125–5131 (2013)
15. Gan, M., Li, H.X., Peng, H.: A variable projection approach for efficient estimation of RBF-ARX model. IEEE Trans. Cybern. **45**(3), 462–471 (2015)
16. Mishra, P.S., Dehuri, S.: Potential indicators for stock index prediction: a perspective. Int. J. Electron. Finance **6**(2), 157–183 (2012)
17. Pandey, T.N., Jagadev, A.K., Choudhury, D., Dehuri, S.: Machine learning–based classifiers ensemble for credit risk assessment. Int. J. Electron. Finance **7**, 227–249 (2013)
18. Huang, G.B., Chen, L., Siew, C.K.: Universal approximation using incremental networks with random hidden computation nodes. IEEE Trans. Neural Netw. **17**(4), 1243–1289 (2006)
19. Yao, J.T., Tan, C.L.: A case study on using neural networks to perform technical forecasting of forex. Neurocomputing **34**, 79–98 (2000)
20. Bebarta, K., Biswal, B., Dash, P.K.: Comparative study of stock market forecasting using different functional link artificial neural networks. Int. J. Data Anal. Tech. Strateg. **4**(4), 398–427 (2012)

Survey on Stock Price Forecasting Using Regression Analysis

Bhupendra Kumar Gupta, Manas Kumar Mallick, and Sarbeswara Hota

Abstract In current days, doing research in stock market is much critical as it shows a nonlinear and random nature based upon several factors. In order to make the profit in future, many invests in stock market rely on some forecast. For prediction of the stock price, people or the investment organization uses some methods and tools. Stock price prediction in stock market is providing main role in stock market business. Use of conventional methods such as fundamental and technical study may not guarantee the consistency of the forecast. In many cases, regression analysis is employed for the forecasting of the stock price. In this paper, we survey the some of the competent regression approach for the price prediction of the stock in stock market. The result of these regression analyses has also been further improvised or can be improvised more using more number of variables and machine learning or data science techniques.

Keywords KDD · ARIMA · Data mining · NSE · NYSE · Ridge · LASSO · Logistic · Elastic Net

1 Introduction

In general, the stock market reflects the economic situation of an economy. If the economic condition of a country is growing, the business firms do higher profits, so the shares of the firms are bought by investors in higher quantity in order to derive the higher dividends. The stock market of the country with growing economic status

B. K. Gupta (✉) · M. K. Mallick
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: bhupendragupta@soa.ac.in

M. K. Mallick
e-mail: mkmallick@soa.ac.in

S. Hota
Department of Computer Application, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: sarbeswarahota@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_16

performs better than the country with lower rate of economic growth. The association between the stock market and the rate of growth of economy may not be ideal, but there must be some correlation, and if the stock market performs well, the economic growth of the country is high, and if stock market falls, then the economic growth of the country is down [1, 2].

Though the economic growth of a country does not always affected by the stock market, it is strongly bonded with how the stock market is behaving. Many studies show that very few % of people of a country participate in stock market because random characteristics of the stock market [2].

Many people have a false impression that investing in stock market is a gamble. Hence, to alter this false impression, it is needed to make them understand about the investment in stock market. Therefore, the forecasting technique plays an important role in the expansion of the investment in stock market by making knowledge for investing in the stock market. The high potential output of the forecasting technique can alter the way of thinking of the people toward stock market. The tools used in data mining too help to forecast upcoming tendency and nature; helping business firms to make knowledge-driven decisions [3, 4]. Knowledge discovery in databases (KDD) and data mining (DM) techniques are competent in extracting the veiled patterns than conventional statistical methods [3–5]. Massive quantity of data are produced in stock market because of which the researchers have to analyze properly and suitably select the technique and apply them for the future price forecasting of the stock.

2 Literature Review

In stock market, generally two basic trading philosophies are followed "Fundamental" and "Technical" approaches. "Fundamental analysis" is done based on the past history and the current business status of the firm while "Technical Analysis" is done based on the trends in price and volume and the news related to the sector in which the firm belongs or any natural calamity. In technical analysis, the timing is important. The forecasting of the stock price is a complex task, and it acts like a chaotic walk and changes rapidly. There are a number of research on it has been conducted of which we are discussing here some.

Adebiyi et al. [6] have discussed a model ARIMA for forecasting the stock price. They have used the dataset of the stock nokia and zenith bank from Nigeria Stock Exchange (NSE) and New York Stock Exchange (NYSE). In this model, author has taken the Bayesian information criteria for determining the best ARIMA model. Hence, the outcome says that using ARIMA model, stock price can be forecasted on short time base.

Ali et al. [7] have done an inclusive study of the essential association between KSE market and macro-economic factors. Shen et al. [8] have suggested the use of data collected from diverse global financial markets with the algorithm based on machine learning for the prediction of the stock index movements.

Chaigusin et al. [9] performed research for forecasting the Stocks Exchange of Thailand (SET) using feed forward back propagation neural networks. They recommended that the right parameters for neural networks model in forecasting of SET are significant. They used a number of factors based on which Thai stock market depends and develops their model for the prediction of Thai stock index. Data with time period of 2003 was used. At the end of the work, three number of network models for the SET index forecast were considered suitable, i.e., three-layer, four-layer, and five-layer neural network with the MAPE of the prediction.

Sheta [10] has developed a model for two nonlinear processes based on fuzzy technique. The development process of the TS fuzzy model was attained in two stages (1) the calculation of the participation functions in the rule antecessor using the formatted input data; (2) the assessment of the outcome factors. The least square method was used by them for the estimation of the parameters. The outcomes were potential.

Kannan et al. in [11] have used methods based on data mining for the discovery of the unseen arrangement using the past data which may contain possible foretelling ability in their investment decisions. The author here has employed five parameters which are very useful in analyzing the equity indices and has obtained the gainful signal. Fazel Zarandi et al. [12] have proposed a expert system for the analysis of stock price based on fuzzy rule. The proposed fuzzy logic-based system allowed to synthesize model rule uncertainties, and each association value of an constituent was interval itself. This fuzzy model is used on fundamental and technical indexes as the input parameters. Lai et al. [13] have synthesized a financial mode based on time series-forecasting by developing and clustering fuzzy decision tree for equity in Taiwan Stock Exchange Corporation (TSEC). The suggested model is integrated with data cluster technique, a fuzzy decision tree (FDT), and genetic algorithms (GA) for the construction of a decision-making scheme using past data and technical indices. In the given model, first the entire past data is classified into k number of subgroups using K-means algorithm. Then, GA was applied to develop the some quantity of terms for each input index in FDT so that model can forecast to most accurate value. The joined actions of stocks in the market are studied by Olaniyi et al. [14].

3 Regression

Regression analysis is a tool used in statistics for estimating the relations between variables. There are several methods in order to model and analyze the variables, but in order to find the relationship between a dependent and more than one independent variables, regression analysis is the better to use. Exclusively, using regression analysis, we can understand the change in typical value of the dependent variable, while any other independent variables change keeping other independent variables unchanged. Mainly, regression analysis approximately calculates the mean value of the dependent variable given the independent variables, i.e., it calculates the normal value of the dependent variable keeping the independent variables constant.

Generally, there are two models used in depicting the relationship between the variables, these are "Deterministic Model" and "Stochastic Model." Regression analysis focuses on stochastic model. In all cases, the estimation target is a mapping of the independent variables known as regression function. The probability distribution function is used in regression analysis to characterize the deviation of the dependent variable about the regression function. For the forecasting purposes, generally regression analysis is extensively used, but many of the times, it is overlapped by machine learning. To determine among all the independent variables which variables are related to the dependent variable and what kind of dependency is there, regression analysis is used. In some of the constraint situation, regression analysis may lead to illusions and provide false relationships, among the dependent and the independent variables. Therefore, carefulness has to be taken, [1] for example, correlation does not imply regression.

There are several methods for regression analyses have been developed. Most common methods are linear regression, and common regression is parametric, in which there are finite number of variables in terms of which the regression function is defined. There may be nonparametric regression in which the regression function stretches out in a precise set of functions, which may be infinite-dimensional. In practice, the data generating technique and the way it is related to the regression approach affects the performance of the regression analysis. As the data generating technique initially are not known, some assumption about the process is taken, and on this, only the performance of the regression depends. Therefore, some of the times when the assumptions are suitable, sufficient quantity of data is available, and in those cases, the performance of the regression analysis is good. If the assumptions for data gathering technique are moderately violated, the performance of regression models though is not optimal but can be employed. But in many situation in which the assumption for data gathering technique is not chosen suitably, regression analysis may leads to the wrong results [2, 3].

More specifically, regression analysis demonstrates the important associations stuck in dependent parameter and independent parameter and the power of impact of several independent parameters on a dependent parameter. Regression analysis can also be used to evaluate the effects of different parameters measured on various scales, such as the changes in price and the some promotional actions. By using these, individual investor/market researchers/data scientists can eliminate and assess the most excellent set of parameters in order to build the forecasting model.

4 Regression Methodologies

4.1 Linear Regression

It is the most basic technique for modeling in which the dependent variable is the continuous variable and the independent variables are the discreet or continuous, and the regression line is a straight line. In linear regression, the relationship between the dependent variable Y and one or more independent variable X is established using best fit straight line called the regression line. Regression line is represented in Eq. (1).

$$Y = A + B * X + \varepsilon, \quad (1)$$

where A is intercept, B is slope of the line, and ε is erroneous term. The above equation can be employed in predicting the aimed variable value depending on the available value of the predictor variable as shown in Fig. 1.

There are several methods used for finding the best fit line for the regression, but among them, most common method used is least square method for the curve fitting which evaluates the best-fit curve by optimizing the total of the squares of the deviations along vertical direction as of every data point to the curve. The positive

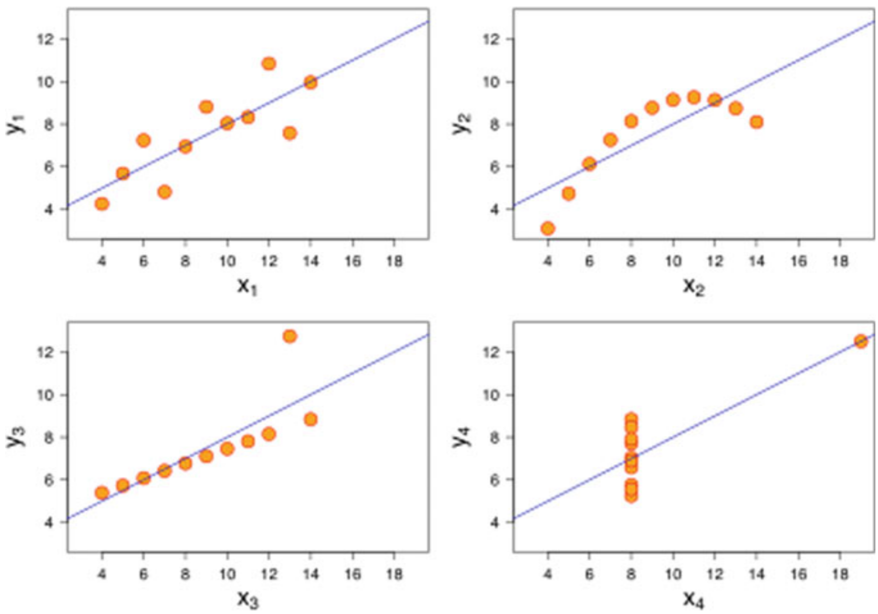


Fig. 1 Relation between dependent and independent variable in linear regression

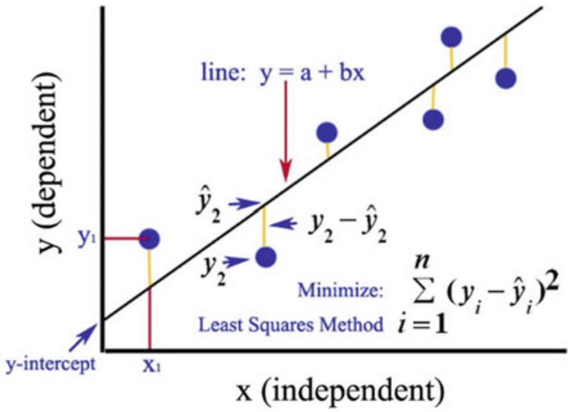


Fig. 2 Curve fitting using least square method

and negative deviations do not cancel each other as the square of the deviations taken first as shown in Fig. 2.

If the independent variable is a single variable, then it is known as single variable linear regression technique, and if there is more than one independent variable, it is known as multi-variable linear regression technique. Multiple regressions experience from multi-co-linearity and auto-correlation. The outliers affect a lot to linear regression. It may horribly influence the regression line and ultimately the forecasted values. One can use step wise approach, forward selection, backward elimination to choose most important independent parameters from multiple independent parameters.

4.2 Polynomial Regression

In polynomial regression, the exponent of independent variable is more than one. In general, the regression equation in polynomial regression is expressed using Eq. (2).

$$a_1x^1 + a_2x^2 + a_3x^3 + \cdots + a_nx^n + b \tag{2}$$

In this regression technique, the best-fit curve is not a straight line. But is any curve that fitted best with the sample points as shown in Fig. 3.

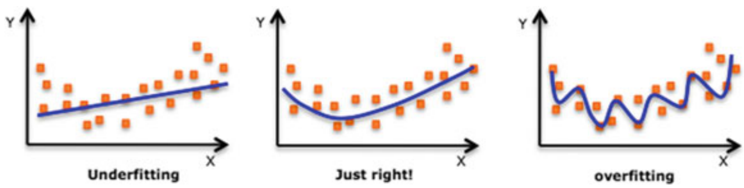


Fig. 3 Polynomial regression

It might happen in order to fit the higher degree polynomial to get lower error, we may get the over fitted curve. In order to avoid over-fitting, additional training samples can be added to avoid the noise learning by the algorithm in the system. A special case of multi-linear regression is the polynomial regression. The forecasting model obtained from polynomial regression can derive accurate features. These features can also be used for classification [3].

4.3 Logistic Regression

At the core of the logistic regression, logistic function is used which is also known as sigmoid function as given in Eq. (3). It was developed by statistician for the description of the population growth in ecology. The domain of the sigmoid function is any real number and gives the output in the range (0, 1) but not equals to 0 or 1.

$$f(p) = \frac{1}{1 + \mathrm{e}^{-p}} \tag{3}$$

where e = natural logarithms base and p = the real value that needs to be transformed.

Logistic regression can be used to find the probability of the event to be success or the probability of the event to be failure. In this regression, the value of the dependent variable lies between 0 and 1. Logistic regression equation is similar to linear regression in which the input values or the independent variables are linearly combined using weighs or the coefficient values to forecast the values of dependent variable. General logistic regression equation having n number of independent variables can be given using Eq. (4).

$$y = \frac{\mathrm{e}^{b_0 + b_1 x + b_2 x^2 + \cdots + b_n x^n}}{1 + \mathrm{e}^{b_0 + b_1 x + b_2 x^2 + \cdots + b_n x^n}} \tag{4}$$

where y is the predicted output, $b_1, b_2, \ldots b_n$ are the coefficient for the independent input values, and b_0 is the bias or intercept term. Each column in the input data has an associated b coefficient (a constant real value) that must be learned from training data. Logistic regression is extensively employed in classification problem.

4.4 Ridge Regression

If the independent variables are highly correlated, i.e., when the data suffered from multi-co-linearity, even though least square estimates are unbiased, their variance is large because of which deviates the observed value large from the true value. Therefore, ridge regression technique is used with biasness which reduces the standard errors. Ridge regression solves the multi-co-linearity problem using shrinkage parameter δ (delta) as shown in the below Eq. (5).

$$\text{MIN}\|X_w - y\|^2 + \delta\|w\|^2 \tag{5}$$

In the ridge regression equation, there are two arguments. First argument is least square expression, and the second argument is δ times the total of w^2 where $w =$ the coefficients. The second term is summed with least square term keeping the aim to minimize the factors to get little variance. All the assumption taken in this regression is same as that of least square method except the normality. Because of which, the value of the coefficient is reduced but not reaches to zero.

4.5 Stepwise Regression

Stepwise regression is employed to set a replica by the addition or the removal of predictor variables through a series of T-tests or F-tests. The parameters which are chosen for the addition or removal are based on the test. Though the technique is beneficial, it has to be used by the skilled researcher who is well known to the statistical test. Like most regression models, stepwise regression model should be watched keenly to notice whether they are useful. In general, stepwise regression requires the addition and removal of the predicators as and when needed.

The variables are added in each step in forward selection with important predicator. The variables are removed in each step in backward elimination with important predictors. The goal of the model is to optimize the forecasting prediction control using less number of predictor parameters. The data set with higher dimension is handled with this regression model.

4.6 LASSO Regression

Least absolute shrinkage and selection operator (LASSO) is same as that of ridge regression. LASSO compensates the absolute size of the regression coefficients and is also competent to reduce the inconsistency and improve the accuracy of linear regression models. LASSO regression equation differs from ridge regression equation as it uses the absolute value instead of the squares of the coefficients as shown in Eq. (6).

$$\text{MIN}\|X_w - y\|^2 + \delta\|w\| \tag{6}$$

In the ridge regression equation, there are two arguments. First argument is least square expression, and the second argument is δ times the total of w where $w =$ the coefficients. Except normality, all other assumption of least squared regression is considered. The features selection in this regression technique is efficient. If the data set duffers from multi-co-linearity, the LASSO regression technique picks only one of them and sets others to zero.

4.7 Elastic Net Regression

The disadvantages of both LASSO regressions are eliminated out by hybridizing it with ridge regression which is known as elastic net regression. The regression equation for the elastic net regression technique is given in Eq. (7).

$$\text{MIN}\|X_w - y\|^2 + \delta_1\|w\|^2 + \delta_2\|w\|. \quad (7)$$

In this equation, we have three components. First one is least square term, and second one is δ times the summation of w^2, and the third one is δ times the summation of w where w is the coefficient. If there are correlated multiple features exist, then elastic net regression is much useful. In case of Lasso, one of these are picked, while both are chosen in elastic net.

5 Conclusions

Here, we have discussed various regression methods used for the forecasting of the price of the stock in a stock exchange. Because of usefulness and needs from the people, data analysis in stock exchange by adding regression technique with other data analysis tool is an active research area. Though a lot of research has been done, existing stock prediction studies still have various limits and restrictions for enhancement. We at last conclude that forecasting the stock price in a stock exchange is very complex assignment and a variety of factors should be taken into consideration to forecast the market more exactly and competently. The goal of our survey is to help the stock brokers and individual investor for understanding different regression technique so that they can use them to some extent and invest their money wisely.

References

1. Fama, E.F.: The behavior of stock market prices. J. Bus. **2**(2), 7–26 (1965)
2. Prasad, D.A.: Security Analysis and Portfolio Management, 3rd edn. I.K. International Publication (2008)
3. Kamley, S., Jaloree, S., Thakur, R.S.: Introduction to Data Mining and Knowledge Discovery, 3rd edn. Two Crows Corporation, Potomac, MD, USA (1999). ISBN 1-892095-02-5
4. Larose, D.T.: Discovering Knowledge in Data: An Introduction to Data Mining. Wiley (2005). ISBN 0-471-66657-2

5. Dunham, M.H., Sridhar, S.: Data Mining: Introductory and Advanced Topics, 1st edn. Pearson Education, New Delhi (2006). ISBN 81-7758-785-4
6. Adebiyi, A.A., Adewuni, A.O., Ayo, C.K.: Stock price prediction using the ARIMA model. In: 2014 16th International Conference on Computer Modelling & Simulation (2014)
7. Ali, I., Rehman, K.U., Yilmaz, A.K., Khan, M.A., Afzal, H.: Causal relationship between macro-economic indicators and stock exchange prices in Pakistan. Afr. J. Bus. Manag. **4**(3), 312 (2010)
8. Shen, S., Jiang, H., Zhang, T.: Stock market forecasting using machine learning algorithms (2012)
9. Chaigusin, S., Chirathamjaree, C., Clayden, J.: The use of neural networks in the prediction of the stock exchange of Thailand (SET) index. In: 2008 IEEE International Conference on Computational Intelligence for Modelling Control & Automation (2008)
10. Sheta, A.: Software effort estimation and stock market prediction using Takagi-Sugeno fuzzy models. In: Proceedings of the IEEE International Conference on Fuzzy Systems, pp. 171–178, Vancouver, BC (2006)
11. Kannan, K.S., Sekar, P.S., Sathik, M.M., Arumugam, P.: Financial stock market forecast using data mining techniques. In: Proceedings of the International Multiconference of Engineers and Computer Scientists (2010)
12. Fazel Zarandi, M.H., Rezaee, B., Turksen, I.B., Neshat, E.: A type-2 fuzzy rule-based expert system model for stock price analysis. Expert Syst. Appl. **36**(1), 139–154 (2009)
13. Lai, R.K., Fan, C.-Y., Huang, W.-H., Chang, P.-C.: Evolving and clustering fuzzy decision tree for financial time series data forecasting. Expert Syst. Appl. **36**(2), 3761–3773 (2009)
14. Olaniyi, S.A.S., Adewole, K.S., Jimoh, R.: Stock trend prediction using regression analysis—a data mining approach. ARPN J. Syst. Softw. **1**(4) (2011)

Intrusion Detection and Classification Using Decision Tree-Based Feature Selection Classifiers

Manas Kumar Nanda and Manas Ranjan Patra

Abstract Feature selection method applied on an intrusion dataset is used to classify the intrusion data as normal or intrusive. Based on the performance evaluation using various feature selection algorithms and the behavior of attributes, we can distinguish the features which plays an important role for detecting intrusions. The dataset has 41 features, out of which some features play significant role in detecting the intrusions, and others do not contribute in the detection process. We have applied different feature selection techniques to extract the predominant feature that are actually effective in detecting intrusions.

Keywords Attribute · Confidentiality · Features · Integrity · Intrusion · Spam

1 Introduction

The massive use of computers and communication through inter connected computer network has widely spread to all over the globe. The interconnection of guided and unguided medium has increased the complexity of network communication. The client–server architecture plays major role on the Internet communication. The increase in usage of the Internet by development using high-speed Internet access devices and with the availability of such sensitive data has caused the attackers to target the servers. The Web servers and Web-based applications are targeted as the most important network-communication devices which serve the client and server-based architectural needs. The Web-based applications are commonly accessible by the ports which are open on firewalls [1]. Nearly about one million of new threats are emerged daily to the global network [2]. Vulnerable of Internet of things (IoT)

M. K. Nanda (✉)
Department of Computer Application, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: manasnanda@soa.ac.in

M. R. Patra
Department of Computer Science, Berhampur University, Berhampur 760007, India
e-mail: mrpatra12@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_17

devices like digital videos and webcams are also used to transfer various malicious and vulnerable software and spams (botnet), like Mirai, is a software which helps to exploit the vulnerability by infecting these IoT devices. Thus, such infected devices may in turn like zombie devices or slaves and which form an army of botnets to attack from multiple different locations to perform a very large scale of distributed denial-of-services (DDoS) attack. Internet is the source of convenient real-time information services like potentially effective threats to deal with the confidentiality, integrity, and availability (CIA), of information, which are also needed to be precisely addressed and with more effective manner and to be permanently handled [3].

2 Intrusion Detection System

The intrusion detection system (IDS) works as an application or monitoring device which identifies some hostile-related activities or as policy violations by the intruder in network. IDS is used to analyze the network traffic and detect some possible intrusive activities in the computer network. Mainly, misuse detection system and anomaly detection system are the two types of intrusion detection systems. It is quite capable of handling and detecting probable attacks from the signatures or well-known patterns and identifying some intrusive activities which deviates from the normal behavior patterns in a monitoring system and can also detect some unknown attacks. The most popular IDSs are like Symantec, McAfee, SourceFire which plays an important role for network monitoring and surveillance, and functions like a network security guard. IDS can also be categorized as a network-based intrusion detection system (NIDS) and host-based intrusion detection system (HIDS) [4]. In NIDS, the intrusion detection system (IDS) is basically installed before and after firewall and to capture network traffic for the entire network segments but the HIDS, in which the intrusion detection system (IDS) is applied on a specific type host to analyze packet information, system logs, and system calls. As compared to NIDS, HIDS is often suitable for the identification of the internal attacks. We have applied a number of techniques to analyze the intrusion data and to build a system which has higher accuracy in detection rate.

3 Data Mining-Based Approach

Most likely data mining techniques are used to discover the way of systematic relationship of data with an approach of determining fundamental information of data. It is broadly divided into two categories such as supervised and unsupervised approach. Classifications and clustering are the best examples of supervised and unsupervised algorithms. In clustering approach, the group of unique objects is mostly based on the characteristics of such data points. Where these data points in a cluster are similar to other data points in the cluster and are dissimilar to the data points in different cluster. By the grouping such type of similar, data points into a cluster which shows the abnormality identification. Hence, this approach may be responsible for the potential increase of false alarm rate. So, the performance of IDS is mainly dependent for the low false alarm rate, which may degrade the performance when it generates high false alarm. Thus, classification is one of the best supervised-based approaches used for classifying the anomalous data or benign, for reducing false alarm rate. It has also the ability to differentiate the unusual pattern of data, which may be suitable for identifying new such attack patterns [5]. Classification is widely used for its very strong ability in the identification of the normal structures with very high accuracy and also contributes toward reducing the false detection [6]. These types of ensemble techniques are used to combine several such classifiers which obtain the better prediction for its accuracy in the performance [7].

3.1 Decision Tree

A decision-based tree is a flowchart of tree-like structure, in which each of the internal node (or non-leaf node) is denoted by a test on an attribute, where each of the branch is represented by an outcome of the test performed, and each of the leaf node (or the terminal node) is held by the class label. The root node is the topmost node of a tree. The decision-based tree induction is about the learning of decision trees from the class labeled training tuples. High-dimensional data can be handled using decision trees. The learning and the classification steps are quite simple and fast for induction of decision tree. The attribute selection methods are used to select the set of attribute that are best suited for partition of the tuples into distinct number of classes, during the tree construction. During the building of decision trees, most of the branches may reflect some noise or cause outliers in the training set of data. This method of identifying and removing the branches with the goal of improving the classification accuracy may be termed as tree pruning which is applied on the unseen data. The basic methods like CART, C4.5, and ID3 adopt the greedy approach (also known as non-backtracking), where decision trees are being constructed by a top-down recursive method using divide-and-conquer approach. Most of the decision tree induction algorithm uses the top-down approach, where it starts with the training set of tuples with the associated class labels. The training set of data is recursively partitioned into a number of smaller subsets so that the tree can be built.

We have applied various decision tree classifiers to classify the KDD CUP dataset. Class for constructing a forest of Random Trees (RF) [8], class for constructing a tree that considers K randomly chosen attributes at each node (RT); K-Nearest Neighbors (KNN) [9], can select appropriate value of K based on cross validation; Decision Tree (J48) is a class for generating a pruned or unpruned C4 (J48) [10]; Projective Adaptive Resonance Theory (PART) [11], uses the mechanism of divide and conquer and builds a partial C4.5 decision tree in each iteration and makes the best leaf into a rule; Multilayer Perceptron (MLP), a classifier that uses backpropagation to classify instances; Sequential Minimal Optimization (SMO) [12] is used to obtain proper probability estimates; Naïve Bayes (NB) [13] Numeric estimator precision values are chosen based on analysis of the training data, the NB Updateable classifier will use a default precision of 0.1 for numeric attributes when build classifier is called with zero training instances [14]; Hoeffding Tree (HT) [15] is an incremental, anytime decision tree algorithm that is capable learning from massive data streams, it exploit the fact that a small sample can often be enough to choose an optimal splitting attribute [16]; KStar (KS) [17] is an instance-based classifier, that is the class of an instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function.

3.2 Feature Selection

A process of feature selection can be used to remove the statistically uncorrelated attributes in the training set with their class labels. To improve the accuracy and efficiency of the algorithm, this reduced set of attributes can be used for classification. The feature selection approach identifies the set of important features for elimination of a number of attributes which do not contribute in intrusion detection process. Detection accuracy is reduced for these irrelevant set of attributes in the intrusion dataset. Using entropy-based ranking methods, like information gain (IG), gain ratio (GR), and symmetrical uncertainty (SU), we can determine and reduce the irrelevant set of features not contributing in the selection process of the dataset.

4 Dataset Used

NSL-KDD dataset is a subset of the KDD 99 [18] dataset, consists of 41 attributes, and used as an intrusion dataset. The redundant records have been eliminated in NSL KDD dataset, and the binary class level contains different types of attacks, namely normal instances, denial of service (DoS), remote to local (R2L), probe, and user to root (U2R). NSL KDD dataset has total 125,973 number of records, out of which 67,343 records are normal data and 58,630 records are malicious data. In this dataset, 41 features are used out of which 38 attributes are numeric and three attributes are symbolic. Each of the attack type is one among the following categories in the number of training and test instances for the experiment.

1. Denial-of-services (DoS) attacks deny services or limit the use of resources by overloading the target system. The different types of such attack are apache, smurf, neptune, ping of death, back, mail bomb, udpstorm, SYN flood, etc.
2. Remote to local (R2L) is an attempt where the attacker send packets to the remote machines over a network which tries to exploit as the vulnerability and an attempt to have an illegal access to the computer for which it does not have a

Table 1 Attack type classification

Class type	Number of instances	Percentage of class occurrences
Normal	67,343	53.48
Denial of services (DoS)	45,927	36.45
Remote to local (R2L)	995	0.78
Probes	11,656	9.25
User to root (U2R)	52	0.04
Total	125,973	100.0

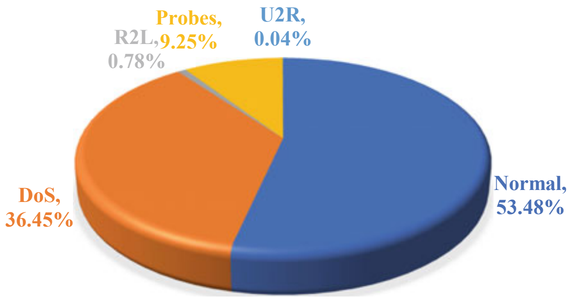


Fig. 1 Percentage of instances of attack type

legal access. The type of such attack is xclock, dictionary, guest_password, phf, sendmail, xsnoop, etc.

3. Probing or surveillance is a type of attack which tries to gather information or knowledge about the physical configuration of a system or a network. The port scans or sweeps a given range of IP address which is in the category of such attack types.
4. User to root (U2R) is a type of attack that attempts to gain access privilege as the root or super_user of a computer in which an attacker has user level access, by this, a non-privileged user tries to gain access to administrative controls or privileges. Such type of attacks is Perl, xterm, etc.

The different attack type with the number of instances is classified and mentioned in Table 1, and the percentage of instances of different class type is shown in Fig. 1.

5 Result Analysis

We have carried out the classification using WEKA tool using a high-end computing environment of core i7 processor, 2.6 GHz, 8 GB RAM, 1 TB hard disk with windows 10 (64 bit) operating system (Table 2).

Table 2 Percentage of instances classified using various algorithms

Class	RF	KNN	J48	PART	MLP	SMO	NB	HT	KS
Normal	53.44	53.33	53.37	53.40	53.09	52.89	42.46	51.95	53.40
DOS	36.45	36.42	36.43	36.44	36.07	35.84	34.58	36.12	36.45
R2L	0.76	0.74	0.75	0.75	0.55	0.60	0.32	0.74	0.76
Probes	9.23	9.21	9.19	9.21	9.15	8.97	6.74	8.93	9.22
U2R	0.03	0.02	0.02	0.03	0.00	0.02	0.04	0.03	0.03
% of correctly classified	99.90	99.72	99.76	99.82	98.86	98.32	84.14	97.77	99.86
% of incorrectly classified	0.10	0.28	0.24	0.18	1.14	1.68	15.86	2.23	0.14
Model building time (s)	7.11	0.05	1.72	3.11	2039.58	17.12	0.23	2.92	0.09
Mean absolute error	0.0035	0.0011	0.0030	0.0024	0.0087	0.2409	0.0607	0.0153	0.0008

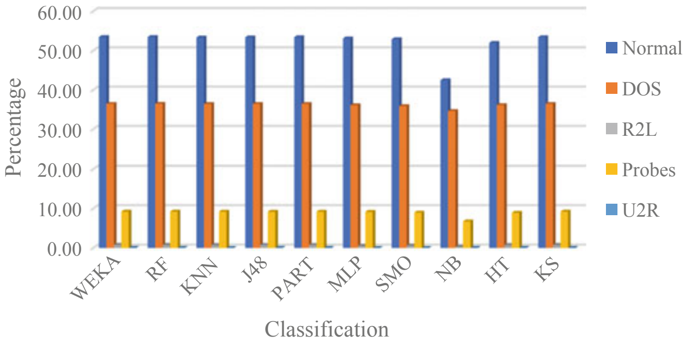


Fig. 2 Percentage of classifications for various attacks

In the classification process using various state art algorithms, we found the random forest (RF) algorithm performs better classification accuracy about 0.10% of instances are not classified. Figure 2 shows the percentage of classification of various attack types (Table 3).

Using entropy-based feature selection method, we have used some efficient algorithms to study the classification of various attack type instances based on their rank. The features are selected randomly to measure the contribution of the features in the classification process. We have listed out some features which do not contribute in the classification process mentioned in Table 4.

On the basis of the rank of the features, the following features are not contributing in the entropy-based feature selection method. The optimum percentage of classification process using the various algorithms is given in Table 3. The optimum set of features applied using entropy-based feature selection gives the best result. The result of the feature selection process is given the respective graph shown. Figure 3 shows the set of features used in the entropy-based feature selection process using random forest algorithm.

Figure 4 shows the set of features used in the entropy-based feature selection process using J48 decision tree-based algorithm.

Figure 5 shows the set of features used in the entropy-based feature selection process using decision table algorithm.

Figure 6 shows the set of features used in the entropy-based feature selection process using projection adaptive resonance theory (PART) algorithm.

6 Conclusion

Most of the researchers have tried to classify the intrusion by applying various data mining and machine learning or hybrid approaches. We have made an attempt to classify and detect the intrusion by applying some data mining tree-based classifiers.

Table 3 Classification of percentage of instances using various attribute selection of rank-based feature selection methods

No. of attribute	Random forest			J48			PART (projective adaptive resonance theory)			SMO (sequential minimal optimization)			(DT) decision table		
	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain
10	99.852	99.431	99.367	99.794	99.392	99.397	99.797	99.404	99.402	96.745	89.243	89.231	99.364	99.163	99.163
12	99.871	99.721	99.781	99.794	99.769	*99.801*	99.832	99.794	99.817	97.032	95.532	96.030	99.316	99.047	99.324
14	99.875	99.795	99.870	*99.801*	99.766	99.792	99.828	99.785	99.817	97.075	95.819	96.912	99.316	99.047	99.324
16	99.893	99.870	99.886	99.772	99.794	99.772	99.827	99.826	99.826	97.146	96.525	97.146	99.324	99.324	99.324
18	99.893	99.864	99.899	99.776	*99.795*	99.778	99.806	99.834	99.806	97.725	96.810	97.725	99.317	99.333	99.317
20	99.893	99.868	99.898	99.763	99.794	99.772	99.824	99.803	99.825	97.782	97.182	97.834	99.317	99.333	99.363
22	99.898	99.880	99.898	99.777	99.787	99.788	99.804	99.826	99.807	97.806	97.318	97.822	99.359	99.333	99.362
24	99.891	99.892	99.895	99.779	99.772	99.781	99.821	99.824	99.820	97.810	97.501	97.809	99.362	99.337	99.362
26	99.904	99.889	99.903	99.782	99.788	99.783	99.821	99.820	99.823	97.901	97.594	97.902	99.362	99.339	99.362
28	99.900	99.896	99.898	99.763	99.780	99.763	99.825	99.817	99.825	98.089	97.951	98.088	99.363	99.339	99.363
30	99.900	99.892	99.898	99.760	99.783	99.761	99.823	99.813	99.821	98.289	98.188	98.288	99.364	99.327	99.362
32	*99.911*	99.906	99.908	99.756	99.766	99.757	99.825	99.824	99.827	98.284	98.264	98.286	99.363	99.343	99.361
34	99.908	99.903	99.905	99.760	99.765	99.761	*99.837*	99.835	*99.839*	98.293	98.314	98.293	99.365	99.367	99.363
36	99.908	*99.916*	99.908	99.762	99.764	99.763	99.821	*99.836*	99.821	98.308	98.315	98.307	*99.370*	*99.367*	*99.367*
38	99.908	99.904	99.908	99.762	99.763	99.763	99.833	99.832	99.826	98.308	98.308	98.308	99.370	99.367	99.367
40	99.909	99.904	99.902	99.762	99.763	99.763	99.816	99.818	99.813	*98.316*	*98.316*	*98.315*	99.369	99.367	99.367
41	99.902	99.906	*99.911*	99.762	99.763	99.763	99.816	99.818	99.813	98.316	98.316	98.315	99.369	99.367	99.367
Max attribute	32	36	41	14	18	12	34	36	34	40	40	40	36	36	36

(continued)

Table 3 (continued)

No. of attribute	Random forest			J48			PART (projective adaptive resonance theory)			SMO (sequential minimal optimization)			(DT) decision table		
	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain
Max	99.911	99.916	99.911	99.801	99.795	99.801	99.837	99.836	99.839	98.316	98.316	98.315	99.370	99.367	99.367
Min	99.852	99.431	99.367	99.756	99.392	99.397	99.797	99.404	99.402	96.745	89.243	89.231	99.316	99.047	99.163
Mean	99.895	99.849	99.861	99.772	99.753	99.750	99.821	99.795	99.796	97.837	97.029	97.330	99.351	99.300	99.342

Table 4 List of attributes does not contribute in rank-based feature selection methods

No. of features	Random forest			J48			PART (projective adaptive resonance theory)			SMO (sequential minimal optimization)			(DT) decision table		
	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain
Optimum at	32	36	*41*	14	18	12	34	36	34	40	40	40	36	36	36
Features do not contribute	f19, f14, f11, f18, f15, f9, f7, f20, f21	f15, f24, f7, f20, f21	–	f12, f37, f36, f32, f24, f31, f40, f2, f41, f27, f1, f28, f10, f22, f8, f13, f16, f17, f19, f14, f11, f18, f15, f9, f7, f20, f21	f34, f31, f27, f33, f23, f9, f2, f32, f41, f36, f18, f17, f28, f1, f40, f13, f16, f19, f15, f24, f7, f20, f21	f34, f37, f33, f23, f36, f32, f31, f40, f2, f27, f41, f24, f1, f28, f10, f22, f8, f13, f16, f17, f14, f19, f11, f18, f15, f9, f7, f20, f21	f11, f18, f15, f9, f7, f20, f21	f15, f24, f7, f20, f21	f11, f18, f15, f9, f7, f20, f21	f21	f21	f21	f15, f9, f7, f20, f21	f15, f24, f7, f20, f21	f15, f9, f7, f20, f21

(continued)

Table 4 (continued)

No. of features	Random forest			J48			PART (projective adaptive resonance theory)			SMO (sequential minimal optimization)			(DT) decision table		
	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain	Info gain	Gain ratio	Symmetric uncertain
Max	99.911	99.916	99.911	99.801	99.795	99.801	99.837	99.836	99.839	98.316	98.316	98.315	99.370	99.367	99.367
Min	99.852	99.431	99.367	99.756	99.392	99.397	99.797	99.404	99.402	96.745	89.243	89.231	99.316	99.047	99.163
Mean	99.895	99.849	99.861	99.772	99.753	99.750	99.821	99.795	99.796	97.837	97.029	97.330	99.351	99.300	99.342

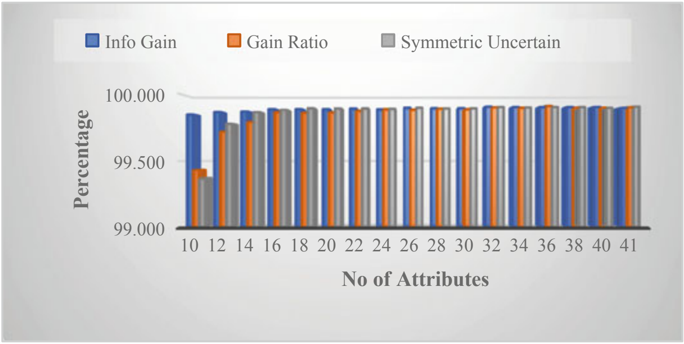


Fig. 3 Attribute classification using random forest algorithm

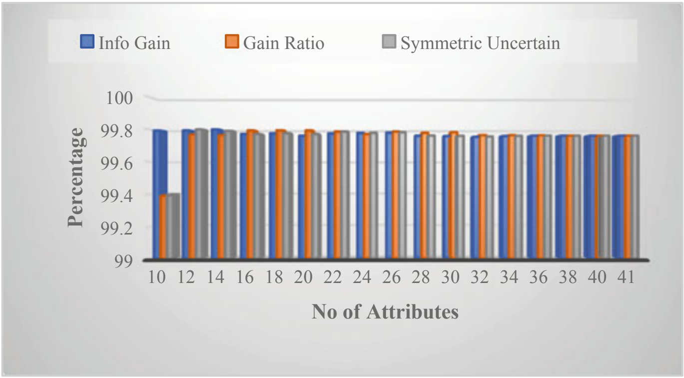


Fig. 4 Attribute classification using J48 algorithm

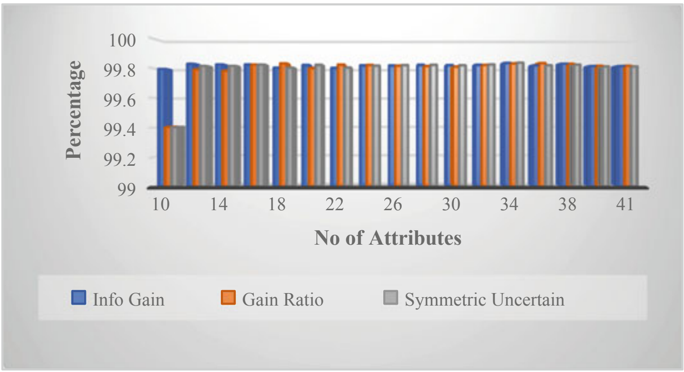


Fig. 5 Attribute classification using (DT) decision table algorithm

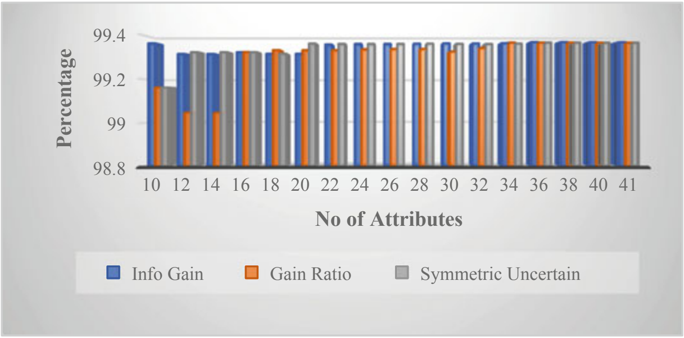


Fig. 6 Attribute classification using projective adaptive resonance theory (PART) algorithm

Most of our attempt has been made to recognize the features that are useful for classifying the intrusion data. More or less there are 11 numbers of features not contributing in the classification process. The entropy-based rank of features is used to classify the intrusion data. The result shows out of 41 features, 11 features rank value is null, and certainly, they do not contribute in the selection process.

References

1. Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., Rajarajan, M.: A survey of intrusion detection techniques in cloud. J. Netw. Comput. **36**(1), 42–57 (2013)
2. Koff, W., Gustafson, P.: CSC leading edge forum data revolution. In: CSC LEADING Edge Forum, p. 68 (2011)
3. Thakare, S.V., Gore, D.V.: Comparative study of CIA. In: Fourth International Conference on Communication Systems and Network Technologies, pp. 713–718 (2014)
4. Muda, Z., Yassin, W., Sulaiman, M.N., Udzir, N.I.: Intrusion detection based on K-means clustering and OneR classification. In: Proceedings of 2011 7th International Conference on Information Assurance and Security IAS 2011, pp. 192–197 (2011)
5. Fiore, U., Palmieri, F., Castiglione, A., De Santis, A.: Network anomaly detection with the restricted Boltzmann machine. Neurocomputing **122**, 13–23 (2013)
6. Muda, Z., Yassin, W., Sulaiman, M.N., Udzir, N.I.: Intrusion detection based on K-Means clustering and Naïve Bayes classification. In: 2011 7th International Conference on Information Technology in Asia, pp. 1–6 (2011)
7. Folino, G., Sabatino, P.: Ensemble based collaborative and distributed intrusion detection systems: a survey. J. Netw. Comput. Appl. **66**, 1–16 (2016)
8. Breman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
9. Aha, D., Kibler, D.: Instance-based learning algorithms. Mach. Learn. **6**, 37–66 (1991)
10. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA (1993)
11. Frank, E., Witten, H.I.: Generating accurate rule sets without global optimization. In: Fifteenth International Conference on Machine Learning, pp. 144–151 (1998)
12. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: Improvements to Platt's SMO algorithm for SVM classifier design. Neural Comput. **13**(3), 637–649 (2001)
13. John, H.G., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338–345 (1995)

14. Kohavi, R.: Scaling up the accuracy of Naïve-Bayes classifiers. A decision-tree hybrid. In: Second International Conference on Knowledge Discovery and Data Mining, pp. 202–207 (1996)
15. Landwehr, N., Hall, M., Frank, E.: Logistic model trees. Mach. Learn. **95**(1–2), 161–205 (2005)
16. Gama, J.: Functional trees. Mach. Learn. **55**(3), 219–250 (2004)
17. Cleary, J.G., Trigg, L.E.: K*: an instance-based learner using an entropic distance measure. In: 12th International Conference on Machine Learning, pp. 108–114 (1995)
18. Adnan, M.N., Islam, M.Z.: Forest PA: constructing a decision forest by penalized attributes used in previous trees. Expert Syst. Appl. **89**, 389–403 (2017)
19. Hulten, G., Spencer, L., Domingos, P.: Mining time-changing data streams. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 97–106 (2001)
20. Perez, J.M., Muguerza, J., Arbelaitz, O., Gurrutxaga, I., Martin, J.I.: Combining multiple class distribution modifies subsamples in a single tree. Pattern Recogn. Lett. **28**(4), 414–422 (2007)
21. Webb, G.: Decision tree grafting from the all-tests-but-one partition. In: IJCAI, San Francisco, CA (1999)
22. Holmes, G., Pfahringer, B., Kirkby, R., Frank, E., Hall, M.: Multiclass alternating decision trees. In: ECML, pp. 161–172 (2001)
23. Islam, G.: Knowledge discovery through SysFor—a systematically developed forest of multiple decision trees. In: Australasian Data Mining Conference (AusDM 11), Ballarat, Australia, pp. 195–204 (2011)
24. Fakhraei, S., Soltanian, H.Z., Fotouhi, F.: Bias and stability of single variable classifiers for feature ranking and selection. Expert Syst. Appl. **41**(15), 6945–6958 (2014)

Role of Cloud Computing for Big Data: A Review

Madhusmita Das and Rasmita Dash

Abstract The application of cloud computing is a recent trend to settle and handle the relevant issues of big data. The term big data can be defined as a dataset, which is too large and complex. The processing of these data is complicated in conventional data processing software. The processing of big data demands large computational infrastructure for data analysis, and this demand can be fulfilled by the integration of cloud computing and big data. Cloud computing is an influential technology to perform gigantic and complex computing. Cloud computing provides hardware and software services through the Internet, so it eliminates the maintenance of costly computing hardware, devoted space, and software. Cloud computing enables big data to control and distribute the stored data in a suitable way. It also provides security to big data through Hadoop. The main idea of big data is to accumulate, handle, visualize, and evaluate the huge amount of data, which is achieved by collaboration with cloud computing.

Keywords Cloud computing · Cloud services · Big data · Hadoop · MapReduce

1 Introduction

Cloud computing revolutionized the usage of computer infrastructure. Characteristics, such as flexibility, elasticity, pay per use, low upfront investment, and a shift of risks, are responsible for the growing development of business infrastructure settings on the platform of cloud computing [1]. The platform of cloud computing can be used to handle and manage big data. If the quantity of data is outside the storage and

M. Das (✉)
Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: madhusmitadas@soa.ac.in

R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: rasmitadash@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_18

processing capabilities of a single machine, then it is called big data. More clearly, if the amount of data is more than hundreds of terabytes, then such data is referred as big data. As the size of the data is too large and the structure is complex, it is difficult to store and process big data in traditional data processing applications [2]. Cloud computing plays an important role to handle the huge storage capacity problem and to provide different flexibility techniques to manage a superlative amount of data [3]. The processing of complex data or big data is achieved using parallel processing, which enhances the performance and scalability of big data. One of the major advantages of big data analysis is that it allows individuals to analyze and recognize threats, which benefit the user to make the data safe and secure. Examples of widely used software systems for big data applications are Google's MapReduce framework and Apache Hadoop [4].

In Sect. 2 of this paper, a background study of the cloud computing with its cloud service model and deployment model is provided. In the same section, the challenges of cloud computing are also discussed. In Sect. 3, the background study of big data, technologies of big data, and its challenges are discussed. Finally, the conclusion is discussed in Sect. 4.

2 Cloud Computing

Cloud is a collection of global servers, which spread through the Internet and are used to store, manage, and process data collectively. Cloud computing uses the Internet as a large-scale computing environment to give huge computing resources to the user application. The major characteristic of cloud computing which make it more popular is multiagency, resource utilization, low cost, scalability, fast provisioning, flexibility, and universal network access. In the present scenario of cloud computing, a user is not required to own all its required resources; instead, the user can use the required recourses from the cloud and pay according to the use. Sometime, users may need some resources temporarily, so in those cases, cloud computing gives a better option to opt for the required resource from the cloud instead of buying it permanently. Cloud computing has flexible architecture, so a large amount of data can be stored from anywhere and also can be accessed from anywhere, depending upon the demand [5]. Cloud computing also supports virtualization, because of this, high-capacity servers are available to different users in parallel. Instead of physical machines, cloud computing provides a virtual machine to the user, which reduces the cost of the resources and offers high storage capacity. Similarly, to achieve scalability and elasticity property of cloud computing, big data analysis in the cloud computing environment is required to explore [6]. So, both the cloud service and skill of data analysis experts require to be managed. Migration of big data to the cloud is a challenging task, which includes data management for the large application through cloud computing and security issues for both cloud computing and big data. There are mainly two type of cloud computing models which are available, such as service model and deployment model. These models provide the platform for big data migration to cloud computing.

2.1 Cloud Service Model

Cloud computing provides services to the user based on their requirements. There are five different cloud service models provided by cloud computing, such as Software as a Service (SaaS), Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Identity as a Service (IaaS), and Network as a Service (NaaS).

Software as a Service (SaaS) In this service model, software is distributed over the cloud through the Internet. Any user can access different software applications and related databases, which are available in the cloud. This service model is also known as "on-demand software application" because the user can access the software application, on his demand and pay according to the use. Some examples of SaaS are Google Docs, Gmail, Salesforce.com, and Online Payroll. All these applications are operating on the cloud remotely, which can be accessed through the Internet. Some of the merits of the SaaS are the administration of the software is trouble-free, compatibility of the software across the company, the association is easy, and accessibility is universal [1, 2].

Infrastructure as a Service (IaaS) In this model, cloud computing provides service in the form of infrastructure such as virtual machines, storage drivers, networks, IP address, and operating system. These entire infrastructures are available in the cloud across the network. Like SaaS, here also the user is not required to purchase the entire infrastructure; rather user can use it on demand and pay accordingly. Here, the user has an environment to run its application in a distributed cloud environment. For example, Flexi scale and Amazon's EC2 are some of the hardware tools, which are available in the cloud, by the service provider. These tools can be used by the end-user according to their requirement [1, 2].

Platform as a Service (PaaS) In this service, cloud provides a platform to the user to use the Web application available in the cloud, without purchasing it. The main difference between SaaS and PaaS is that SaaS provides software over the cloud to the end-user, but PaaS gives a platform in the cloud over the Web to create software. For the end-user, who needs to design software, the required platform such as operating system, programming language, and Web servers are provided by PaaS. Examples of PaaS services are engine of Google and Force.com, Windows Azure, AppFog, Openshift, and VMware Cloud Foundry [1, 2].

Identity as a Service (IaaS) This service provides cloud computing user's identity information as a digital entity. The service reduces the problem of remembering username and password combinations [1, 2].

Network as a Service (NaaS) This service is also based on pay per use concept. It allows the user to use virtual network infrastructure provided by service providers and pay accordingly. The NaaS also maintain and manage network recourses, which lead to less workload to the user [1, 2].

All the above models receive popularity in the current situation, because of its hazard free provision of different services. The benefit of cloud computing can be categorized based on the user's requirement. In the next subsection, different types of deployment models of the cloud are discussed.

2.2 Cloud Deployment Model

The deployment model describes the nature and purpose of the cloud uses. It is majorly categorized into four different categories, which are discussed below.

Public Cloud Model In this model, the service of the cloud is available to the user publically. It means the service provided by the cloud is available to the user without any third-party involvement. Some of the examples of public cloud facilities are IBM, Google, Amazon, Microsoft, etc. In all of these examples, any user can use the service provided by them publically [3, 5].

Private Cloud Model In this model, a particular organization or client constructs their local cloud. This local cloud is known as a private cloud, as it is dedicated to a particular organization or to a user, who built it. The private cloud is accessible and maintained by the organization that creates it. The cloud service is not available to any general user like in public cloud; rather private cloud permits only the authorized users of the owner organization to use the private cloud platform. The organization that has dynamic, critical, secured, management demand requirement should use private cloud [3, 5].

Community Cloud Model In a community cloud model, a set of organizations, who are having the same computing concern, built a community cloud model. The participated organization may belong to the same community. The community model is also maintained by the same community, who built it. This model is used whenever joint business organizations, ventures, or research organizations want to share their resources among themselves [3, 5].

Hybrid Cloud Model Hybrid cloud model can be a combination of private, public, or community cloud model. Sometimes an organization may need to access non-critical services, from the public cloud and critical service like organizational data handling from private cloud. The same organization also want to distribute their resources among other joint organization, in this case, a hybrid cloud model is suitable. The hybrid cloud model gets the benefit of all other models. This model also overcomes the limitations of rest three models [3, 5].

2.3 Challenges and Issues of Cloud Computing

Some of the challenges and issues of cloud computing are categorized with respect to data, information handling, and storage.

Availability/Accessibility In the cloud environment, resources or data are available and accessible to authorized users through the Web. Availability of resources becomes an issue, whenever more request is there to the cloud for the resources within a stipulated time. As the number of clouds authorized user increase, the availability and accessibility become a major issue as the speed of resources transfer is degraded. The challenge of cloud computing is to deliver the resources to the requested user with high-quality service.

Reliability The issues of reliability of cloud computing are in an aspect of performance, connectivity, and security. The reliability of performance arises whenever there are several users in the public cloud model. Similarly in the noisy neighbor environment, reliability of connectivity and security is also an issue. These issues must be addressed by the user level.

Interoperability Interoperability refers to an understanding of public and private cloud services with each other's APIs, configuration, data formats. It becomes an issue when a customer shifts from one cloud to another cloud.

Portability In cloud computing, portability is of two types, application portability, and data portability. Application portability is the capability to shift an application from one cloud to another cloud. This becomes a challenging issue at the time of application move from one cloud to other cloud environments. Similarly, data portability means the capability to shift data between different clouds [6].

Performance Performance is one of the major components of decision making, on whether to adopt a cloud computing platform by a user or not.

Security and Privacy All the essential benefits of cloud computing are eliminated if the security aspects is not fulfilled. As the base of cloud computing is to share resources among users through cloud over a network, the security of resources becomes a challenging task [6]. Similarly, privacy is another challenging aspect as cloud computing infrastructure is based on resource sharing.

Transformation Transformation refers to the transfer of data to a suitable form. This is the biggest challenge in migration of big data to cloud computing.

3 Big Data

Nowadays, Internet, online service, and stock market are widely used by different business organizations. The use of such technologies results in an exponential increase of data and information flow. Big data can handle and process this huge amount of data and information. Big data is a term that refers to a huge amount of data, and these data may be of different verities, such as structure, semi-structured, or unstructured. Big data does not worried about the amount of data, rather it considers how wisely the data can be used so that some of the benefits can be achieved like cost reduction, time reduction, new product development with optimized effort, better decision making, etc. Some of the examples of big data fields are stock exchange data, social media data, transport data, search engine data, etc. In all these fields, data are generated in a huge amount that cannot be stored and processed in a single local physical device. Cloud computing is one of the solutions for big data storage and management. The data collected from a various sources are of different verities, and such data need to be analyzed properly. The analysis of big data is a process to uncover hidden patterns and useful information. This information can be used by the business organization for its business benefits. Hadoop and MapReduce are some of the technology used for big data analysis. These technologies support large dataset processing in the cloud. Big data become popular with its five 'V' term, they are volume, velocity, variety, veracity, and value [7, 8].

Volume Different organization gathers information from different sources, such as from business transaction, data from social media, login data, and also from machine to machine data. As the volume of the data goes on increasing, it may lead to problems related to storage and process, but because of the technology advent with big data, the trouble of voluminous data got decreased.

Velocity It defines the speed of data transfer. As the volume of data is more, it may degrade the velocity. The technology of sensor and smart metering gives the solution to the fast data transfer requirement.

Variety In big data, the data are from various sources and also in different formats. The data can be of structured, unstructured, semi-structured, numeric data, traditional data, text data, audio data, video data, login data, financial data, and even encrypted data. These varieties of data need to managed and processed with conformity. Big data mechanism provided a solution to this issue.

Veracity The varieties of data which are collected have various quality and accuracy. The exposure of such data should be done with properness. Big data achieves the required timeliness of data at the time of exposure.

Value Big data is the collection of different data of various variety, these data must be analyzed and processed. However, after processing the value of the data should not be changed. Big data ensures the correct value of the data after processing, and this processed data can be used for the betterment of the business. Big data also manages the complexity of multiple sources data, cleans it, and transfers to the requisite without changing its value.

All the issues related to the above five 'V' are answerable in big data technology. For the processing of the big data, parallelization techniques are adopted, because of which better performance and scalability are achieved. The computation of big data is not possible in a local machine, and similarly, storage is also not possible in a single physical device, so big data is migrated to cloud computing, which fulfills the required multiple processing environments and multiple storage environment for the huge amount of data. In big data technology, data are distributed in various servers, which are connected through a network and can be accessed by the Web. The cloud service provider (CSP), provides data to the user according to their requirement. The CSP also maintains the integrity, authentication, and recovery of data [8]. Cloud computing provides some of the basic advantages to big data, with respect to the speed of data transfer and the capacity of storage. Another advantage of big data is that it allows its user to visualize the data, which helps in finding new business opportunities. Some features, which make the big data popular among different organizations, are third-party auditor, encryption-based storing of data, privacy-preserving public auditing, nonlinear authentication, secure, and dependable storage. All these features of big data are implemented in cloud computing through various technologies.

3.1 Technologies Supported by Big Data

Some of the technologies which are supported by big data are discussed in the bellow section.

Column-Oriented Database The traditional row-oriented database has high update speed but the performance is degraded, in case if the volume of the data increases and also if the data are unstructured. The column-oriented database is used to store the data with columns, as a substitute of rows. It is useful for a huge volume of data as well as for unstructured data. It has also a very fast query processing time. The main disadvantage of this model is that they do only updates in batches, so the update time is much slower in comparison with other models.

Schema-less Database The schema-less database is also known as the NoSQL database. This type of database is focusing on a large volume of data along with various varieties, such as structured, unstructured, semi-structured, and even structure. The requirement of big data processing is achieved in this type of database. The conventional limitations such as read-write consistency and distributed processing are overcome.

MapReduce It is a framework that handles a large amount of data processing in a consistent and fault-tolerant manner. This framework divides the data into several chunks. These chunks are processed by map jobs in parallel manner. A file system is used to store the input and output data. The map steps and reduce steps are used to distribute the data. The output of the map step is sorted and then sent to the reduce step. The above process helps to resolve large data problems. A query is formed to access the related data. These data can be further reduced. The cloud service providers use the MapReduce framework to process the huge amount of data. This framework is based on master-slave architecture [9].

Hadoop It is the implementation of the MapReduce framework. Hadoop is an open-source platform for managing big data. It is developed by Apache software foundation and used by various business organizations such as Amazon, Facebook, Intel, Apple, and Microsoft. This helps to analyze and process huge volume of data such as weather data, traffic sensors data, social media, and intermediate data of machine. Hadoop created clusters to distribute data among different machines in the cloud. To manage and operate such a huge volume of data, Hadoop Distributed File System (HDFS) is used [9].

Hive It is a "SQL-like" bridge, which allows a traditional business logic application to execute queries against the Hadoop cluster. Hive is developed by Facebook, but nowadays it became open source.

PIG PIG is another bridge just like Hive, which brings Hadoop closer to the business user and developers. PIG consists of "Perl–like" language. PIG also allowed query execution over the data stored in the Hadoop cluster. It was developed by Yahoo and made a fully open-source like Hive.

3.2 Challenges in Big Data

The amount of generated data goes on increasing day by day. Managing this vast amount of data is the biggest challenge in IT industries.

Availability The availability of data to the user becomes an issue, as the data are distributed on different clouds. The performance of accessing data and transferring this data into the required form is a big challenge in big data technology [10].

Security Security is a challenging task in big data technology, as sensitive data are accessible from the cloud, and it is vulnerable to hackers. Big data require multilevel security.

Scalability The data speed and CPU speed is different. Due to this mismatch of speed, the large volume of data cannot be transferred to the processor with in the required time. To maintain the scalability, cloud computing should provide service to the user in the required time, and this becomes an issue in big data technology [11].

Big Storage Another important challenge of big data is that it requires a big storage system in the cloud to store the vast amount of data. Data can be of various forms like text, image, audio, video, etc. These various forms of data are used by different mediums like mobile, sensor technology, social media, radars, etc. Hence, Storing varieties of vast amount of data become a tough task [12].

4 Conclusion

In this paper, overview of cloud computing, services of cloud computing, and its challenges are discussed. The big data and its supported technology with its challenges are also highlighted. Nowaday's big data becomes a rising area of research. The integration of big data and cloud computing serves the requirement of many organizations and companies. The usage of big data over cloud computing is an exploring area for many researchers and IT professionals. For the advancement of cloud features and big data features, technology like Hadoop and MapReduce is the best solution.

References

1. Jaiswal, S., Patel, S.C., Singh, R.S.: Secured outsourcing data & computation to the untrusted cloud—new trend. CSI Commun. **38**(12) (2015)
2. Acharya, D.P., Kauser, A.P.: A survey on big data analytics: challenges, open research issue and tools. Int. J. Adv. Comput. Sci. Appl. **7**(2) (2016)
3. Chandrashekar, R., Kala, M., Mane, D.: Integration of big data in cloud computing environment for enhanced data processing capabilities. Int. J. Eng. Res. Gen. Sci. **3**(2) (2015)
4. Srinivas, J., Venkata, K., Reddy, S., Qyser, A.M.: Cloud computing basics. Int. J. Adv. Res. Comput. Commun. Eng. **1**(5) (2012)
5. Raja, J., Ramakrishna, M.: A comprehensive study on big data security and integrity over cloud storage. Indian J. Sci. Technol. **9**(40) (2016)
6. Singh, S., Rathi, V.K., Chaudhary, B.: Big data and cloud computing: challenges and opportunities. Int. J. Innov. Eng. Technol. **5**(4) (2015)
7. Kaur, I., Kaur, N., Ummat, A., Kaur, J., Kaur, N.: Research paper on big data and Hadoop. Int. J. Comput. Sci. Technol. **7940** (2016)
8. Hassan, R., Manzoor, R., Ahmad, M.S.: Big data and Hadoop: a survey. Int. J. Comput. Sci. Mob. Comput. **6**(7) (2017)
9. Harini, S., Jothika, K., Jayashree, K.: A review of big data computing and cloud. Int. J. Pure Appl. Math. **118**(18) (2018)
10. Inukollu, V.N., Arsi, S., Ravuri, S.R.: Security issues associated with big data in cloud computing. Int. J. Netw. Secur. Appl. (IJNSA) **6**(3) (2014)
11. Reddy, V.A., Reddy, G.R.: Study and analysis of big data in cloud computing. Int. J. Adv. Res. Comput. Sci. Manag. Stud. **3**(6) (2015)
12. Venkatesh, H., Perur, S.D., Jalihal, N.: A study on use of big data in cloud computing environment. Int. J. Comput. Sci. Inf. Technol. **6**(3) (2015)

Role of Cloud Computing in Geotechnical Engineering

Rashmisikha Behera and Manas Ranjan Das

Abstract Geotechnical engineers, nowadays, are able to examine many problems in much greater depth with the help of advanced hardware and software. Again, the engineers have started to depend less on sophisticated software and hardware with the availability of cloud computing services. Here, only the cloud computing system interface software needs to be run simply like a web browser. In view of the inherent risk and variability associated with geotechnical engineering, the geotechnical modelling tools usually resort to statistical and numerical techniques to take care of uncertainties in the key problem parameters. With the advent of cloud computing, geotechnical engineering needs computer technology not only for analysis and mathematical modelling, but also for recording, storing, retrieving, processing, visualizing and displaying of important geotechnical data much efficiently. The present paper broadly reviews the role of cloud computing as well as system identification and parameter identification integrated with cloud computing in geotechnical engineering.

Keywords Cloud computing · Geotechnical engineering · System identification · Parameter identification · Inverse analysis

1 Introduction

An important role is played by index and engineering properties of soils and rock mass in geotechnical engineering. Inherent variability of soils and rock mass leads to uncertainties with regard to onsite geological conditions, material behavior and overall response including soil structure interaction. The deterministic approach, on which geotechnical design is traditionally based, requires representative properties of

R. Behera (✉) · M. R. Das
Department of Civil Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: rashmisikhabehera@soa.ac.in

M. R. Das
e-mail: manasdas@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*, Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_19

materials in each identified zone and single analysis adopting a large factor of safety which is uneconomical. Another solution to this problem is a stochastic or probabilistic framework within which performance of geo-structures can be expressed. In this framework, material properties of each zone can be expressed as a probability density function using all the field data. These data may then be used in assessing structural performance leading to probabilistic definition of response from which reliability-based factors of safety and characteristic values may be obtained. Over the past few years, more attention is paid to inverse analysis to determine the properties of soils and rock mass from field measurements. It becomes very difficult to work with too complex constitutive model despite the fact that several experimental means are available for determining soil and rock parameters. This difficulty provokes uncertainties in the input parameters. A good number of parameter identification methods called inverse analysis have been proposed over past few years in order to overcome these difficulties. Parameter identification in geotechnical engineering is a typical complicated nonlinear function optimization problem. Cloud computing-based system identification for identifying model parameters is also called parameter identification. System identification is the process of modelling an unknown system based on a set of input–output. It has several applications in geotechnical engineering, such as determining constitutive model parameters by using observations in laboratory tests [16], supported excavation [11, 17], tunneling, in situ testing [24]. System identification for identifying model parameters is also called as parameter identification.

This paper reviews the role of cloud computing in geotechnical engineering either directly or as an integrated part of system identification or parameter identification. Organization of the paper is as follows. Section 1 broadly introduces to requirement of cloud computing in geotechnical engineering. Section 2 describes the direct role of cloud computing in geotechnical engineering. Section 3 highlights the integration of cloud computing with parameter identification forming a system identification architecture and its role in geotechnical engineering. A review on inverse analysis in geotechnical engineering along with the general formulation of parameter identification process is placed in Sect. 4. Application of cloud computing-based parameter identification to modeling of stone column, a specific problem in geotechnical engineering is highlighted in the same section. Scope with some challenges is placed in Sect. 5.

2 Direct Role of Cloud Computing in Geotechnical Engineering

2.1 State-of-the-Art Cloud Computing

There are four different cloud models based on type of requirements. They are (1) private cloud where same organization governs, owns and operates on computing resources, (2) community cloud where both community and organization are served by computing resources, (3) public cloud where one government organization governs, owns and operates on computing resources, (4) hybrid cloud where different clouds bind together the computing resources. Similarly, cloud computing offerings can be classified into three categories, such as (1) Software as a Service (SaaS) where a vendor or a service provider hosts the applications and makes it available to customer over a network, (2) Platform as a Service (PaaS) where the user accesses the service hosted in cloud via the Internet, (3) Infrastructure as a Service (IaaS) where a virtualized environment on the Internet gives access to computing resources. Cloud computing comprises of two components: front end and back end. The interfaces and applications required to access the cloud computing platform on the client side of cloud computing system constitutes the front end. The cloud itself consisting of the resources required for cloud computing services like data storage, security mechanisms, virtual machines, servers, etc., refers to back end. Cloud computing is enabled by a technology called virtualization. By this technology, a single physical server is partitioned into multiple logical servers. Each logical server performs like a physical server where an operating system and application can be run independently.

2.2 State of the Art Approaches for Geotechnical Cloud Computing

2.2.1 Geotechnical Data Transfer

Data transfer format is made up of two components: (1) geotechnical data dictionary and (2) rules creating the file. Two cloud engines have been developed: (1) Association of geotechnical and geo-environmental specialists (AGS) launched first version of its AGS format in 1992. (2) Data interchange for geotechnical and geo-environmental specialists (DIGGS) launched its first version in 2005.

Both are supported by Keynetix, Cloud since 1998 and 2005, respectively. A database can produce an AGS file (based on CSV) or DIGGS file (based on XML) or JSON file as long as all the data to be transferred is in the database. Any system used to store and retrieve geotechnical data should be DIGGS and AGS compatible. If it can read and write these formats, then whoever is gathering or using the data will be able to send or receive it in these formats. The technology exists in the form of cloud-based computing.

2.2.2 Geologs

It is a database management system, in the field of geotechnical and geo-environmental engineering, developed and robustly tested by relevant engineers and programmers for the intended purpose. It is a cloud software written using latest programming languages and is in operation since 2007. As a project develops, Geologs create borehole, trial pits and rotary logs. Integrated contamination feature enables it to create contamination analysis report. All of these are done in a cloud where technical requirements are use of a browser and reasonable Internet speed. This cloud engine reduces the time spent by the engineers on non-expert tasks and enhances efficiency.

2.2.3 Geo-computing Clouds

Geoscientific computing is performed based on this type of cloud service. Linux and Windows platforms are integrated seamlessly enabling workflows in geotechnical engineering with RiVA Nimbus. Remote visualization servers (RVSs) provide geotechnical workstations which are separated into several series in order to address the requirements of wide variety. Customization of a private cloud can be done by selecting RVS in any combination as follows.

X-series: Highly demanding workloads can be addressed by this series through quality performance. The most powerful Intel and nVidia processors available can be employed by these workstations containing large memory.
H-series: These workstations exhibit high performance through optimization. Four 256 GB virtual workstations are hosted by each RVS with 12 GB vGPU profile per virtual workstation. Work flows like seismic interpretations in geotechnical engineering can be performed suitably by these workstations.
P-series: These workstations can bring a good balance between performance and density. Eight 128 GB virtual workstations are hosted by each RVS with a 2 GB vGPU profile per virtual workstation. Moderate geotechnical workflows can be performed suitably by these workstations.

2.2.4 Other Geotechnical Cloud Engines

Apart from these three major cloud engines already discussed, there are some other cloud engines performing in geotechnical engineering such as Landmark DGS365, Schlumberger Blucube, Amazon Web Services, Microsoft Azure. Performances of these engines are moderately comparable to that of geo-computing cloud.

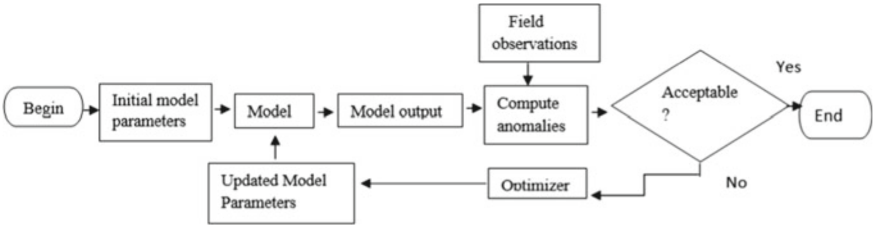


Fig. 1 Flowchart depicting system identification

3 Cloud Computing and Parameter Identification Integrated in System Identification

3.1 System Identification

The flowchart describing the system identification in general is given in Fig. 1. Generation of unknown soil model parameters randomly or from past experience is performed first. Next, the computation of the models with given parameters is carried out to produce outputs. Popularly, the chosen models are finite element models. Outputs thus obtained are then compared with field observations. Quantification of anomalies is performed by objective functions resulting in objective function values. Very small values ensure system identification and obtaining unknown parameters. Else, the optimizer tries minimization of the values and updating of the model parameters. The process loops back till satisfaction.

Multiple users can be served at the same time through large-scale capabilities of cloud computing developed by IT technologies implemented by such a system. Incorporation of Google engine like V8 Java script and implementation of multi-objective optimizer enables the system to be extensible and greatly flexible making it general purpose.

3.2 Component of SI

Cloud computing and parameter identification (PI) are the two main components of software architecture of system identification.

3.2.1 Cloud Computing

This consists of a result database, web front end and back end. Again, back end consists of node managers, a task queue and a shared file system. Thus, the cloud computing infrastructure is formed for a system. Mell and Grance [15] have underlined the virtues of cloud computing, namely measured service, pooling of resources, access to network and auto-service on demand. Task archive is uploaded using PHP web pages which is hosted by Apache web server and stored in shared file system. Then, task queue stores a new task and gets implemented in MongoDB as a collection. MongoDB with its GridFS capability helps implementation of share file system. Then, node managers, in a cloud computing facility, play an important role in polling the task queue to locate unexecuted task, if any. Then, node manager fetches each unexecuted task and sends it to task queue which in turn hands over it to PI component where it is executed.

3.2.2 Parameter Identification

The software architecture shown in Fig. 2 has two parts; left component refers to the cloud computing, and right component describes the parameter identification (PI) process. The PI component consists of task executor, optimizer, A/O converter and analyzer. Connections between optimizer, analyzer and A/O converter are established by the task executor, and then, the optimizer conducts the PI process after the control is transferred to it. Defined analyses are run by the analyzer being invoked by optimizer during the process. A/O converter computes the objective values after being invoked by A/O converter at the end of an analysis. These objective values are stored in result database and implemented using MongoDB.

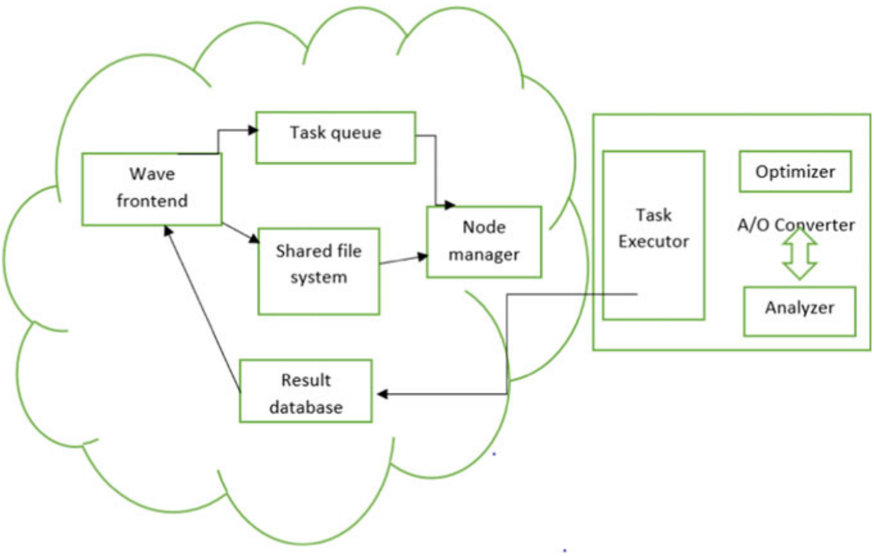


Fig. 2 Software architecture of system identification

4 Inverse Analysis and Parameter Identification Supported by Cloud Computing

4.1 General Formulation of Parameter Identification Process

A type of constrained nonlinear optimization problem is solved during the process of parameter identification. Mathematically, it can be expressed as:

$$\begin{aligned}&\text{Min } f\,(z)\\&\text{Subject to}\\&(\text{Constraint Equations}),\\&Z \in D_z\end{aligned} \tag{1}$$

where

f objective function to be minimized
Z vector of parameters
D_z admissible vector of parameters.

Constraint equations may consist of equality and inequality constraints or may be a system equation similar to finite element formulation. Thus, an optimizer plays very important role in parameter identification process in geotechnical engineering. More than one type of laboratory tests is often required to determine the model parameters. For example, five parameters have been identified for modified cam clay (MCC) model, out of which two parameters which are the main deformability parameters have been derived from oedometer test and the remaining three parameters have been derived from triaxial compression test. Similarly, the model developed for S-clay used 8 parameters, that for hardening soil used 9 parameters and that for MIT-SI used 16 parameters. All parameters for a particular model have been derived from different kinds of laboratory tests. Thus, the optimizer in parameter identification process in geotechnical engineering may need to handle multiple objectives.

Different types of optimization techniques have been used in solving geotechnical problems. These optimization techniques are classified into two categories: gradient-type method and direct search method. Application of optimization techniques has a great contribution towards solving geotechnical engineering problems [1–3, 7, 12]. Use of optimization technique in solving various geotechnical engineering problems increased specially in analyzing stability problems. But due to their inefficiency in handling complex problems, need of new efficient methods like particle swarm optimization, ant colony optimization, simulated annealing and genetic algorithm (GA) were observed. Application of GA in geotechnical engineering was reported in early nineties [18] which mainly involve stability problems like slope stability analysis [8] and problems regarding reinforced soil slope. Still there exist very limited application in problems related to ground improvement and parameter identification.

Limited application of multi-objective optimization in geotechnical engineering is reported recently [9, 10]. Deb and Dhar [9] have used multi-objective optimization as a tool for parameter estimation of reinforced embankment on soft soil, with one objective only. Some of the specific stability problems like ground improvement techniques, bearing capacity of foundation, load carrying capacity of stone columns, etc., where the use of multi-objective optimization need to be explored.

4.2 Inverse Analysis and Parameter Identification

Direct identification of the constitutive parameters of the soil layers is very difficult to perform in majority field tests in geotechnical engineering. This uses roughly estimated values of engineering and index properties of soils and rocks. Such a scenario puts a strong limitation in using finite element method for design of geotechnical structures. In this context, parameters of soil constitutive models need to be identified by inverse analysis. Again, a particular type of geotechnical structure requires a particular parameter identification method suitable for the purpose.

Trial values of unknown parameters obtained from the database of a cloud computing engine are used as input values in the chosen finite element code (e.g., ABACUS, PLAXIS, ANSYS) for simulation of the real problem till the difference between measured values and numerical results is minimized. Thus, the problem is reduced to parameter optimization which effectively takes care of the limitations in using the finite element method concerning the soil model in geotechnical engineering. Inverse analysis technique was first introduced by Gioda and Sakurai (1987) in the geotechnical engineering to identify the elastic properties of in situ rock masses. With the arrival of sophisticated software with high configuration computers and cloud computing technology, inverse analysis has found wide application in geotechnical engineering with success. A fast-growing attraction towards parameter identification strategies and optimization algorithm is observed to make the inverse analysis procedures in geotechnical engineering automated [5, 4, 13, 14, 19, 24].

The advantage of using inverse analysis along with the genetic algorithm as an optimization method is to provide a proper range of material properties of the soil in various zones of the dam out of measured data, without any need for implementing any destructive method. However, engineering judgment is essential to consider for the final evaluation of model parameters. Cloud computing integrated with inverse analysis forms a powerful tool in geotechnical engineering.

4.2.1 Application in Earth and Rockfill Dam

Values of soil parameters are very difficult to be determined in case of earth and rockfill dam. Destructive soil exploration methods, particularly in the impervious zone of the dam, are usually not recommended from dam safety point of view. In the absence of sufficient reliable data of material properties, construction of a constitutive model becomes difficult. In such a scenario, various researchers have resorted to inverse analysis for identification of soil parameters for earth and rockfill dam.

Vahdati et al. [21] explored the effectiveness of the inverse analysis technique in constructing a Mohr–Coulomb model for one earth and rockfill dam. They used a case study of 45 m high earth and rock fill dam with 6.5 m crest and downstream slope of 1.85 horizontal to 1 vertical as well as upstream slope of 1.71 horizontal to 1 vertical. The dam consists of an impervious core of moraine materials encompassed by fine and coarse filter founded on a blasted rockfill. Its first stabilizing berm was built in 1990, and second stabilizing berm was built in 1993. Average error function was minimized using genetic algorithm-based optimizer. They found the inverse analysis technique to be effective, but quite expensive. In such a case, cheaper cloud computing technology can prove to be quite useful.

Vahdati et al. [22] tried to explore the possibility of applying inverse analysis technique for constructing a hardening soil model for the same earth and rock fill dam. Chosen parameters for the model referred to various reservoir water levels in the first case and number of berms built in the second case. Search engine chosen for optimization was genetic algorithm. Search domain was reduced from a starting larger one to a reduced smaller one, and its impact on the solution was studied in both the cases. Field data was obtained from inclinometer measurements of horizontal displacements, etc. The topology of the average error function, i.e., the objective function, was found to depend on chosen model, type of search engine and size of search domain. A cloud computing engine, in this context, can be a cutting-edge tool.

Toromanovic et al. [20] employed inverse analysis on a strengthening model of an earth and rockfill dam in Northern Sweden to identify the parameters. They also used an error function as objective function and genetic algorithm as search engine in the optimizer Measured data was obtained from field monitoring. The technique was found to be effective and can prove to be quite powerful when integrated with cloud computing engine.

4.2.2 Application in Underground Cavern and Tunnel

In depth excavation of tunnels and underground caverns, neutralizes tension in the soil which in turn causes plastic and elastic deformations in the area of such excavations as well as ground surface settlement. In these cases, material behavior, identification of relevant parameters and choice of a suitable soil model greatly influence the output of numerical analysis.

Çelik [25] conducted a study on tunneling using two models, namely—Mohr–Coulomb model and Hardening soil model. Close to accurate predictions of ground movement in case of underground excavations in urban areas bears utmost importance. Çelik observed a more accurate representation of actual behavior by hardening soil model than that yielded by Mohr–Coulomb model due to a responsible parameter like stiffness. In view of so much significance of a database of material behavior, cloud computing supported parameter identification can facilitate application of inverse analysis.

Xiang et al. [23] applied parameter identification technique to tunneling. They also analyzed the identifiability and reliability problems considering initial damage parameters for damage model constructed for jointed rock mass. This process was a constrained nonlinear optimization problem. In view of the complex and heterogeneous ground condition, an advanced tool like cloud computing which can support inverse analysis leading to parameter identification seems quite relevant. This technology may reduce the cost involved too.

4.2.3 Application in Modeling of Stone Columns

Soft soils for foundation purposes need improvement in terms of bearing capacity, final settlement, degree of consolidation, etc. Stone columns provide a solution to this problem. They help in improving the engineering properties of naturally available soft soil, such as—permeability, shear strength and stiffness. Also, in loose granular soils, stone columns take care of liquefaction problems through enhancement of relative density and stiffness. But a lot of uncertainty is associated with the installation of stone columns. When soil gets extremely soft, uncertainty in geometric shape, stability, continuity makes the stone column unsuitable for the intended purpose. Stone columns are never installed in single units. They are installed in groups. All these make modeling of stone columns very complicated. Moreover, availability of reliable data on material properties of stone columns is not so easy. These properties are not even measured for each project. These properties even vary along the length of individual stone columns and depend on many factors. Important material parameters influencing performance of stone columns like elastic modulus and friction angle of stone column vary with confining pressure. Value of friction angle obtained from box shear test is different from that obtained from triaxial compression test. Plastic strains prevalent in columns affect the peak values of friction angles appreciably. Unit weight of gravels is another parameter which exhibit large variations. Lengths of the column and its diameter as well as spacing are some other important parameters which need to be identified properly. Application of inverse analysis in the modeling of stone columns is very rarely reported. In view of these complexities associated with stone column performance, cloud computing-based parameter identification can simplify the complicated modeling of stone columns.

5 Research Scope and Challenges

In the field of geotechnical engineering, strength and stability of soil plays a vital role. A lot of parameters are there which have direct and indirect influence on both the strength and stability of the soil. So, it is not easy to identify the parameters accurately. Many researchers have used different techniques for parameter identification, in which basically attainment of experimental data is done followed by application of numerical techniques. But it is seen that parameter identification with a limited number of experimental data may have low accuracy, while to get a large set of experimental data by personally performing all the requisite test is quite a herculean task. In order to solve this problem, an information technology-based solution is proposed, i.e., to use cloud computing as a tool for obtaining large set of data stored in geotechnical data engines. Carter et al. [6] explained that geotechnical design is much more than just analysis, and it comprises of data gathering prior to analysis, as well as observation and monitoring during and following construction.

When it comes to challenge of the present work, it can be said that challenges in cloud include lack of resources and expertise in the field for a new author. Skill is required to work with cloud engines. Security is a great concern, and increased security of data is considered to have potential benefit, but it may create some problems in accessing the required data. So, in order to form a generalized model for solving various geotechnical problems, large amount of data accumulation is required which can be done easily through cloud computing. Novelty of the present study lies in the fact that a completely unexplored area like cloud computing-based parameter identification in geotechnical engineering can open up avenues for research.

6 Conclusion

An attempt has been made in the present paper to explore the possibility of extensive application of a cutting-edge tool like cloud computing in geotechnical engineering. There is a lot of variability and uncertainties associated with the mechanical behavior of materials in geotechnical engineering. Geotechnical modeling is also very complex. In this scenario, proper identification of materials properties bears utmost importance. Recently, inverse analysis technique for parameter identification has gained popularity. The fact that availability of reliability data from field tests or from laboratory experiments is very poor makes the inverse analysis technique much more relevant. The issue of result database, so barely needed in geotechnical engineering, can be solved by cloud computing technology with less cost being involved. Thus, cloud computing-based parameter identification can prove to be a powerful tool in geotechnical engineering. An attempt has also been made in the present work to highlight the cloud computing engines available in the market specifically for geotechnical engineering. Different geotechnical applications have been demonstrated where the role of cloud computing has been tried to be underlined. Future scope lies in the fact that a lot of stochastic approaches in geotechnical engineering can be transformed into deterministic approaches with the help of a cutting-edge tool like cloud computing coupled with parameter identification.

References

1. Basudhar, P.K., Valsangkar, A.J., Madhav, M.R.: Optimal lower bound of passive earth pressure using finite elements and nonlinear programming. Int. J. Numer. Anal. Methods Geomech. **3**, 367–379 (1979)
2. Basudhar, P.K., Madhav, M.R., Valsangkar, A.J.: Optimization techniques in bearing capacity analysis. Indian Geotech. J. **8**(2), 105–110 (1978)
3. Basudhar, P.K., Valsangkar, A.J., Madhav, M.R.: Nonlinear programming in automated slope stability analysis. Indian Geotech. J. **9**(3), 212–219 (1979)
4. Calvello, M., Finno, R.J.: Selecting parameters to optimize in model calibration by inverse analysis. Comput. Geotech. **31**(5), 410–424 (2004)
5. Calvello, M., Finno, R.: Calibration of soil models by inverse analysis. In: Numerical Models in Geomechanics NUMOG VIII, pp. 107–116 (2002)
6. Carter, J., Desai, C., Potts, D., Schweiger, H., Sloan, S.: Computing and computer modelling in geotechnical engineering, pp. 1157–1252 (2000)
7. Chen, W.F., Snitbhan, N., Fang, H.Y.: Stability of slopes in anisotropic nonhomogeneous soils. Can. Geotech. J. **12**, 146–152 (1975)
8. Das, S.K.: Slope stability analysis using genetic algorithm. Electr. J. Geotech. Eng. **1** (2005)
9. Deb, K., Dhar, A.: Optimum design of stone column-improved soft soil using multi-objective optimization technique. Comput. Geotech. **38**(1), 50–57 (2011)
10. Deb, K., Dhar, A., Bhagat, P.: Evolutionary approach for optimal stability analysis of geosynthetic-reinforced stone column-supported embankments on clay. KSCE J. Civ. Eng. **16**(7), 1185–1192 (2012)
11. Finno, R., Calvello, M.: Supported excavations: observational method and inverse modeling. J. Geotech. Geoenviron. Eng. **131**(7), 826–836 (2005)
12. Krugman, P.K., Krizek, R.J.: Stability charts for Inhomogeneous soil condition. Geotech. Eng. J. South East Asian Soc. Soil Eng. **4**, 1–13 (1973)
13. Lecampion, B., Constantinescu, A.: Sensitivity analysis for parameter identification in quasi-static poroelasticity. Int. J. Numer. Anal. Methods Geomech. **29**(2), 163–185 (2005)
14. Lecampion, B., Constantinescu, A., Nguyen Minh, D.: Parameter identification for lined tunnels in a viscoplastic medium. Int. J. Numer. Anal. Methods Geomech. **26**(12), 1191–1211 (2002)
15. Mell, P., Grance, T.: The NIST Definition of Cloud Computing, p. 7. Special Publication 800-145. National Institute of Standards and Technology, U.S. Department of Commerce (2011)
16. Mulia, A.: Identification of soil constitutive soil model parameters using multi-objective particle swarming optimization. Master thesis, National Taiwan University of Science and Technology, Taipei, Taiwan (2012)
17. Rechea, C., Levasseur, S., Finno, R.: Inverse analysis techniques for parameter identification in simulation of excavation support systems. Comput. Geotech. **35**(3), 331–345 (2008)
18. Simpson, A.R., Priest, S.D.: The application of genetic algorithms to optimization problem in geotechnics. Comput. Geotech. **15**(1), 1–19 (1993)
19. Tarantola, A.: Inverse Problem Theory. Elsevier, Amsterdam (1987)
20. Toromanovic, J., Mattsson, H., Knutsson, S., Laue, J.: Use of parameters identified by inverse analysis in an earth- and rockfill dam application (2017)
21. Vahdati, P., Levasseur, S., Mattsson, H., Knutsson, S.: Inverse Mohr-Coulomb soil parameter identification of an earth and rockfill dam by genetic algorithm optimization. Electron. J. Geotech. Eng. **18**, 5419–5440 (2013)

22. Vahdati, P., Levasseur, S., Mattsson, H., Knutsson, S.: Inverse hardening soil parameter identification of an earth and rockfill dam by genetic algorithm optimization. Electron. J. Geotech. Eng. **19**, 3327–3349 (2014)
23. Xiang, Z., Swoboda, G., Cen, Z.: Parameter identification and its application in tunneling (2003). https://doi.org/10.1007/978-3-7091-6099-2_8
24. Zentar, R., Hicher, P.Y., Moulin, G.: Identification of soil parameters by inverse analysis. Comput. Geotech. **28**(2), 129–144 (2001)
25. Çelik, S.: Comparison of Mohr-Coulomb and hardening soil models numerical estimation of ground surface settlement caused by tunneling (2017)

Energy-Efficient Clustering with Rotational Supporter in WSN

Priyambada Parida, Bandita Sahu, Ami Kumar Parida, and Sambit Kumar Mishra

Abstract The wireless sensor network is an evergreen field of research. Everywhere we are using the sensor. Since the sensors are small in size and have less amount of initial energy, the energy saving becomes highly important and challenging. Wherever we deploy these sensors, it may or may not be accessible all the time. Hence, these should be implemented with a suitable algorithm to utilize energy efficiently. We have proposed an energy-saving algorithm by reducing the overheads of the cluster head (CH). In order to assist the CH, an assistant is selected called the supporting CH (SCH). Generally, this responsibility is rotational. Most of the nodes get a chance to serve CH so that the energy utilization is uniform. Through the proposed algorithm, the lifetime of the network in creased. This proposed algorithm is simulated using NS3 simulator and proves the energy-efficient clustering and increased lifetime as compared to other algorithms without the use of SCH.

Keywords Cluster head · Non-cluster head · Supporting cluster head · Over head · Energy usage · WSN

1 Introduction

Sensor devices are used for detecting and responding to various signals from the soundings. These are referred to as nodes in a network. As its name, it is used to sense

P. Parida (✉) · B. Sahu · A. K. Parida
GIET University, Gunupur, Odisha, India
e-mail: priyamabadaparida@giet.edu

B. Sahu
e-mail: banditasahu@giet.edu

A. K. Parida
e-mail: amiparida@giet.edu

S. K. Mishra
Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha 751030, India
e-mail: sambitmishra@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_20

data from the environment. For conversion of one form of signal to another, these sensors are used. Parameters converted by sensors include temperature, pressure, humidity, speed, and many more. They are used to keep a record of the sensed data to identify any changes in the environment. Sensors provide a response signal for the changes they observed. Several types of sensors are available such as radar sensor, IR sensor, weight sensor, motion sensor, temperature sensor, chemical sensor, and many more. The WSN is a network where many sensors are connected as a node to measure the physical parameter of the environment, such as temperature, pressure, humidity [1]. The main applications of WSN are to monitor and surveillance in the battlefield, to monitor air pressure, temperature, humidity level, noise level, patient diagnoses and monitoring, agriculture, etc. If we have a different type of data, then we categories those data into a number of groups. The method of defining a similar kind of nodes among a group of nodes is called clustering [2]. Clustering is an important method to extend the lifetime of WSN. It connects a group of sensing devices with a leader node called a cluster head (CH) [3]. Except for the cluster head, other nodes are called cluster member, and they are directly connected with the CH. It receives the sensed data from its member and transmits it to the base station (BS). Several algorithms have developed in this field to take care of many things. The research is an ongoing process in terms of increasing the battery life, energy consumption [4], processing overheads of the nodes, and lifetime [5] of the network.

In this paper, we have proposed a method for efficient energy utilization. An assistant to the CH is appointed or selected to reduce the processing of CH. Generally, the assistants are rotational, so it produces uniform energy utilization of the follower nodes. By reducing the processing overhead of CH, its lifetime and energy saving is increased. The followers who satisfy the criteria to become an assistant get a chance to serve the CH. This paper is organized into five sections. Section 1 contains the introduction part. In Sect. 2, we have described various works related to WSN. Section 3 represents a detailed description of the proposed approach with the system model and various algorithms. In Sect. 4, we have shown the result obtained from the simulation. The conclusion and the future works of the proposed method are mentioned in Sect. 5.

2 Related Works

The authors in [6] have designed a new algorithm for making the WSN fault-tolerant. It is an on-demand-based algorithm. Once a CH gets failed or could does not meet the CH criteria, a new CH is selected. Each time, a supporting CH is selected with the CH and NCH. The clustering is done with respect to the signal to noise ratio (SNR) [7] of the sensor nodes. For the avoidance of data loss, the data to be sent is copied in the SCH. When the CH gets failed, SCH [8] performs the task of CH. The authors have concentrated on dynamic clustering and fault tolerance. Underground wireless sensor network (UWSN) [9] is suggested with robust architecture and energy efficiency. By using UWSN, we can lessen energy usage. Here, different nodes are divided into groups known as clusters. If a node in inactive mode or node has no work, then it consumes less power. This method can implement the concept of fault tolerance with time constraint [10]. There is an approach called a selective data transmission approach by which processed data is send through NCH. NCH sends the new sensed data to the CH; otherwise, it is kept in its input queue. If the input queue is free, then it first processes the existing queue data and stores the new data. Here, only the selected dates are transmitted and also it reduces the cluster overhead. Data transmission makes less by employing on-demand data transmission [11] method where data transmission begins when CH demand; otherwise, it will make the node inactive. Hence, the frequency of transmission and energy utilization is reduced. In clustered underwater sensor network (CUWSN) [12], a less number of data transmissions use less energy and extend the network lifetime. Here, the author proposed a new protocol called as selective data transmission in SNR-based cluster (SCSD) [13].

In this method, SNR takes an important role. With the help of SNR, value clusters are organized, and cluster heads (CH) are selected. By selecting the data to be transmitted, the number of transmissions between cluster heads and its follower nodes decreased. The congestion problem can also be avoided. WSN is restricted with energy [14], the computational power of the network, and memory capacity. An inspector node is selected to reduce the probability of CH selfishness [15]. It takes care of the operation done by CH rather than assisting it. In the recent past, there are so many problems arising regarding more battery power consumption. Therefore, the author recommended for getting productive architecture with energy efficiency in WSN for collection and aggregation [16] of data. Benefit in WSN technology is the utilization of low-cost sensors which are used for sensing many things such as physical and environmental conditions, data processing, and wireless communication. In WSN, there are some restrictions like transmission range of sensor nodes is limited, limited in storage and processing capability, and limited in energy resources. Triple umpiring system (TUS) [17] proved that the performance in a wireless sensor network is better. The lifetime of WSN is increased by using clustering technique [18]. Here, the authors modified the ad hoc on-demand distance vector routing (AODV) [19] by using signal-to-noise ratio (SNR)-based dynamic clustering. This method gives an efficient and secure routing protocol for WSN by using SN- based dynamic clustering process. Here, it separates the nodes into clusters and selects one cluster head among the nodes. Except for CH, the other nodes are known as non-cluster head (NCH), and the NCH are connected with CH-based SNR values.

3 Proposed Method

In the proposed method, we have designed a new system model consisting of cluster head (CH), non-cluster head (NCH), base station (BS) and supporting cluster head (SCH). This method is proposed to reduce the processing overhead of CH. Always a CH is assigned with an assistant called as supporting cluster head or SCH. The SCH

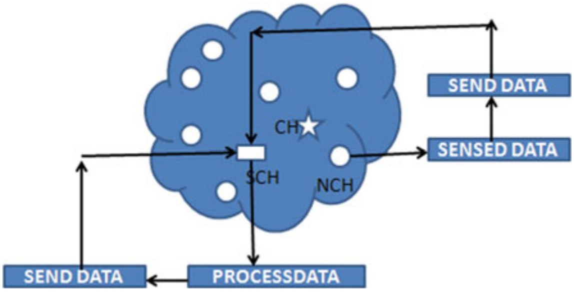


Fig. 1 System model

receives data from all the NCH in the respective cluster, processes it, and sends the processed data to the CH. Less energy is required for the CH for data processing and data transmission. However, energy utilization is more at SCH. Once the SCH fails to satisfy the energy constraint, the charge of assistance is assigned to another node. The new node is called as SCH now. This assistance is performed on a rotational basis. Each and every NCH present in the cluster may get a chance to serve an SCH that system model shows in Fig. 1. This process can be accomplished using four steps. The first step clustering is performed. Selection of CH, NCH, and SCH are made in the next subsequent steps. The steps are (1) clustering of nodes, (2) CH selection, (3) NCH selection, and (4) SCH selection. The description of each steps are as follows.

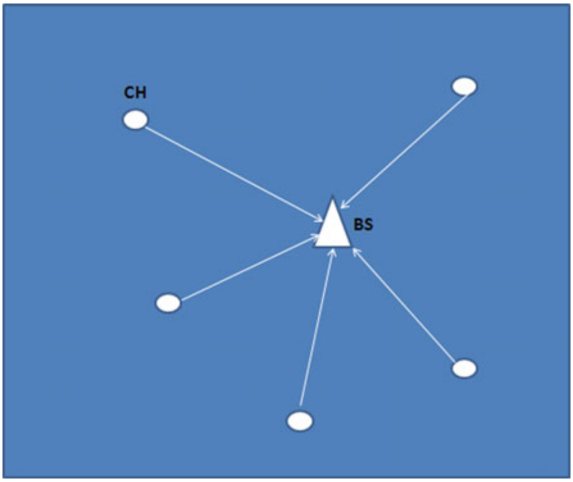
3.1 Step 1—Clustering of Nodes

We are classifying the nodes into various groups. Each group is called a cluster. This classification is done based on signal-to-noise ratio (SNR) of nodes with respect to BS. The nodes with high SNR values are selected as the head or leader of a group. A number of clusters in approach are assumed as 5.

3.2 Step 2—Selection of CH

In this algorithm, we find the CH depending on SNR value. Here, '*n*' is defined as the number of nodes and '*k*' is known as the number of clusters. When '*i*' value is less than or equal to the number of nodes, then we get SNR values. When '*j*' value is less or equal to the number of the clusters, then we get the maximum SNR value of *i*. After that, we able find which node will be cluster head. The CH selection algorithm is mentioned below. Figure 2 shows the selected CH with respect to the SNR values.

Fig. 2 *k* number of CH selections



Algorithm 1: CH_Selection (*n*, *k*)

1. $i = 1, j = 1$
2. while($i \leq n$) // n number of nodes
3. get_SNR[i]
4. while($j \leq k$) // k number of clusters
5. get_maxSNR[i]
6. CH[j] = Node[i]
7. $n = n - k$

Algorithm 2: NCH_Selection (*n*, *k*)

1. $i = 1, j = 1$, CH[i] = { }
2. while($i \leq n$)
   for ($j = 1; j \leq k; j$++)
   get_distance [i][j]
3. get_mindistance [i][j]
   CH[i] = NCH[i][j]

3.3 Step 3—Selection of NCH

In this step, we are selecting the NCH or the follower nodes in each cluster. Nodes having lesser SNR value as compared to the CH can become the NCH. Depending upon the distance of the nodes, each CH selects their followers. The distance of each node is computed from each CH. The CH with minimum distance is considered as its CH. And the corresponding node is assigned to the selected CH as the NCH. The NCH selection algorithm is mentioned below. The graphical representation of NCH selection for each cluster is shown in Fig. 3.

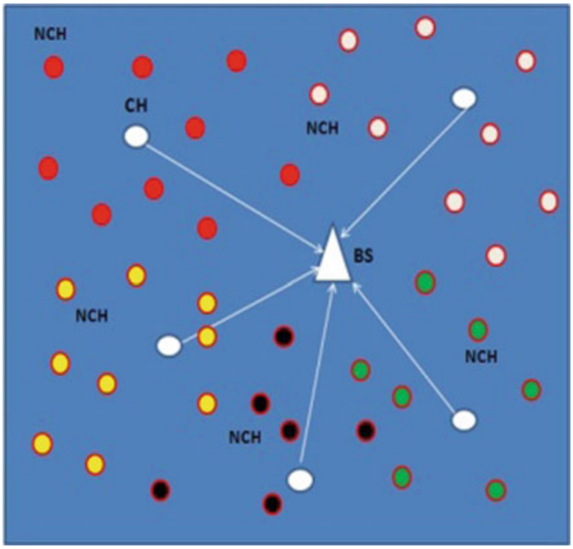


Fig. 3 Selection of NCH

3.4 Step 4—Selection of SCH

In this step, the supporting CH is selected to reduce the processing overhead of CH. The SCH acts as the assistant to the CH. Processing of the received data items is completed at the CH. Each NCH in the cluster sends the sensed data to the SCH instead of sending it to the CH directly. As an assistant, the SCH processes the data and sends it to the CH. It reduces the processing overhead, energy consumption, and life span of CH. The SCH works for a fixed number of iterations. For becoming an SCH, each node has to satisfy the upper bound of the threshold value.

Table 1 Assumption parameters

Parameters	Value
Area of sensing	100 × 100
Number of nodes	50
Number of clusters	5
Topology	Random
Initial energy	2.24 J
Threshold energy	1.85 J
Transmission power	2 W
Sensing range	10 m

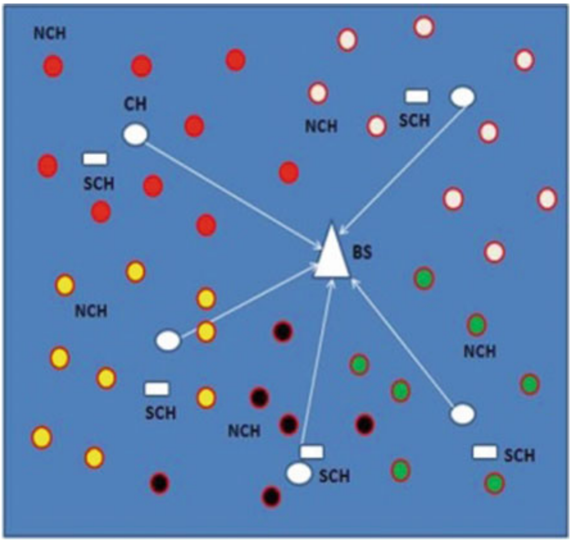


Fig. 4 Selection of SCH

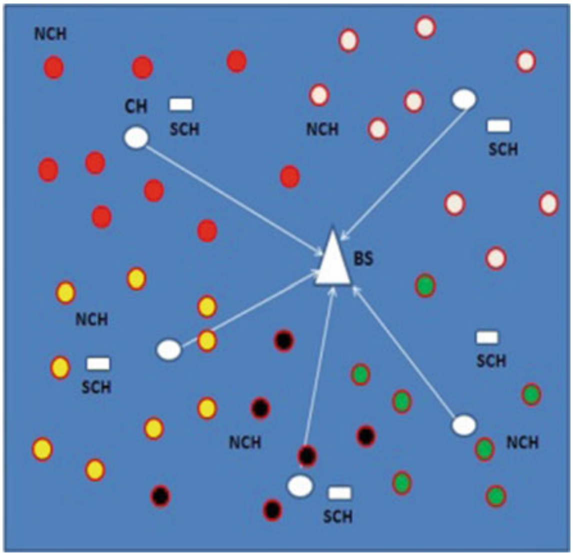


Fig. 5 SCH after some iteration

The threshold energy is mentioned in Table 1. However, the upper bound is set to threshold energy plus 0.5. The selected SCHS is shown in Fig. 4. Similarly, Fig. 5 shows the position of SCH after some iteration. The algorithm is as follows.

Algorithm 3: SCH_Selection (*k*)

1. for $i = 1$ to n
2. for $j = 1$ to k
3. if (node energy[i] > £th && distance (node[i], CH[i]) == Extract node is)
   a. SCH[i] = node[i]
   b. Status (SCH[i] = 's')
4. else
   get next node[i] with minimum distance from CH[j]
5. Return SCH[j]

4 Simulation

By using the random topology, 50 nodes are deployed is a 100 × 100 grid. The position of the base station is assumed to be fixed. It is simulated using NS3 simulator. The assumption parameters are mentioned in Table 1. SNR values are computed with respect to the base station and are shown in Fig. 6. The cluster heads are selected, and the processing overhead is computed for each. As SCH is taking care of data receiving and data processing, CH has less overhead. Figure 7 represents the CH overhead with the use of SCH and without the use of SCH. As the CH performing less number of operations, it utilizes less energy. CH only performs the data receiving from SCH and data send to BS. Hence, it lasts for a longer period of time and saves more energy as compared to other algorithms. However, if the number of NCHs increases in a node, it affects the overhead of SCH rather than CH. Figures 8, 9, and 10 represent the lifetime, energy saving, and the number of iterations the CHs and SCHs are active, respectively. The SCH is rotational in nature. After some iteration, a new SCH is formed. The selection of new SCH is based on energy it is having. Thereby, each node gets a chance to assist the CH. The energy usage is uniform in the network

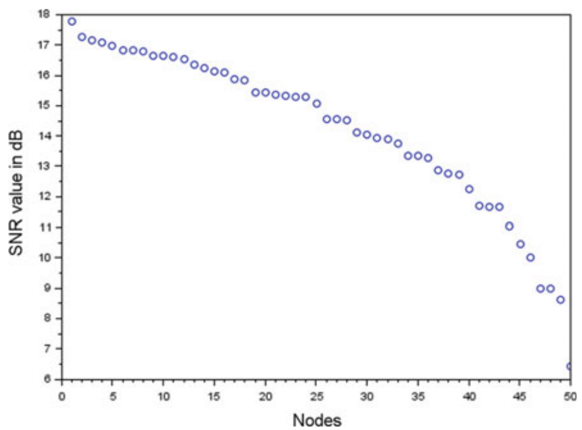


Fig. 6 SNR value of nodes with respect to BS

Fig. 7 CH's overhead

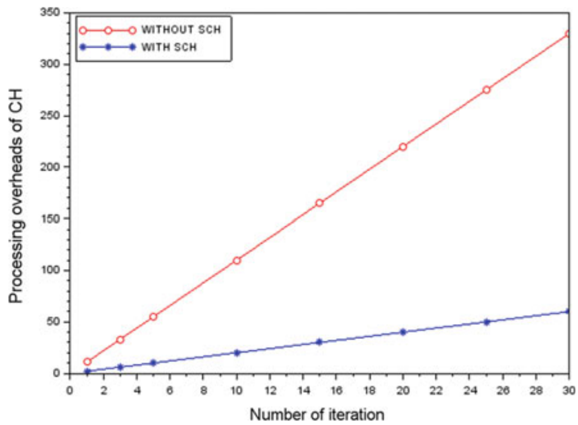
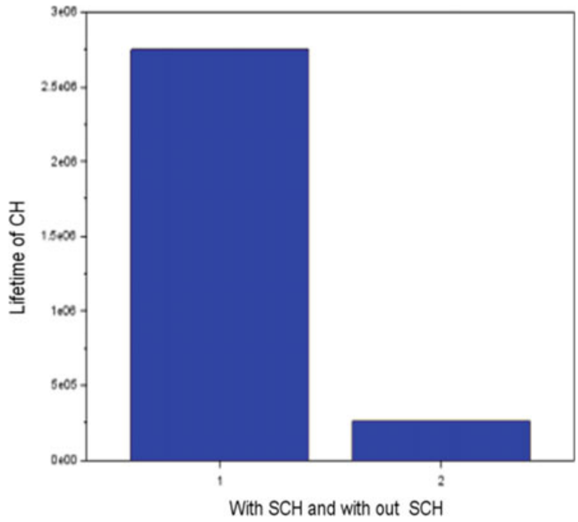


Fig. 8 Lifetime of CH with respect to iteration



[20]. And it runs for a longer period of time. This rotational nature of SCH increases the lifetime of the sensor network deployed, as shown in Fig. 11. The algorithm complexity is measured in terms of the asymptotic notations. The algorithm is a linear equation with the number of nodes. However, total computation is more than that of using fixed SCH and without using SCH. The complexity is shown in Fig. 12. The authors [7] have formed the cluster using SNR values. However, the CH selection is a complex process. Unlike CH selection in LEACH, it also required two phases. It is an efficient routing protocol named as efficient and secure routing protocol for wireless sensor networks through SNR-based dynamic clustering (ESRPSDC). More amount of energy is required and thereby the network lifetime is less as compared to the proposed algorithm. Figures 13 and 14 show the supremacy of the proposed approach.

Fig. 9 Energy usage

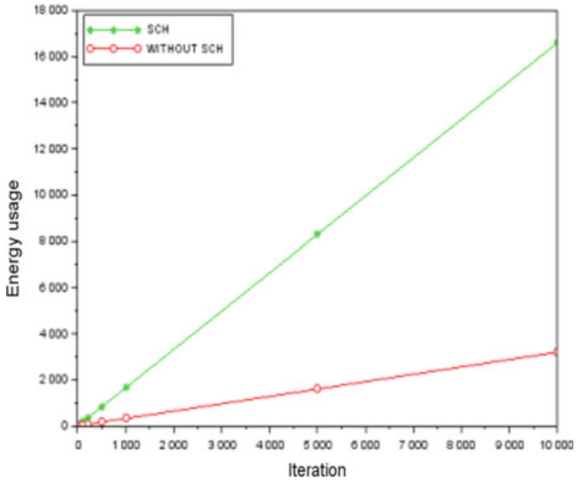
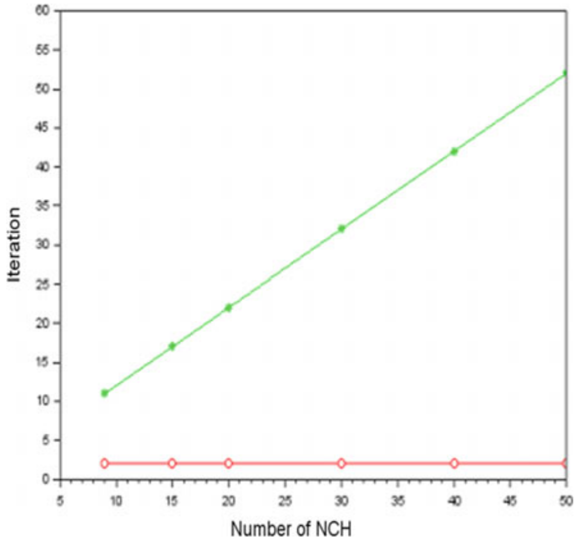


Fig. 10 Effects of increasing NCH in number of iterations



5 Conclusion

The reduction of the processing overhead of the cluster head is assisted by selecting a new node. The assistant is the supporting CH (SCH). In this paper, we have introduced the concept of rotational SCH. It reduces the energy utilization, thereby increasing the lifetime of the network. The CH performs less number of operations and hence active for more number of iterations. We have only concentrated on the energy usage and processing overhead of the nodes in this paper. In the future, we will work on data security and robustness of the network.

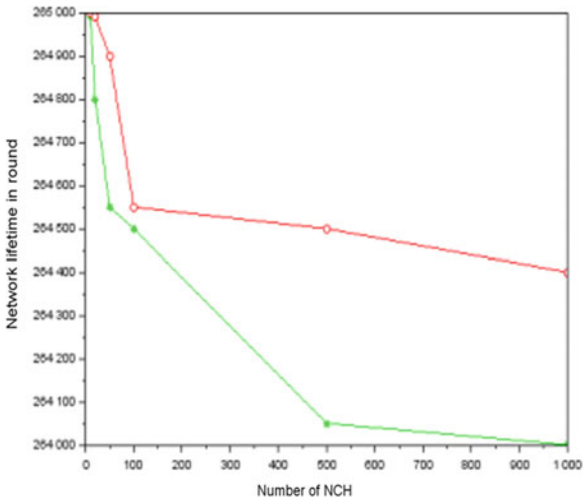


Fig. 11 Network lifetime due to uniform energy usage

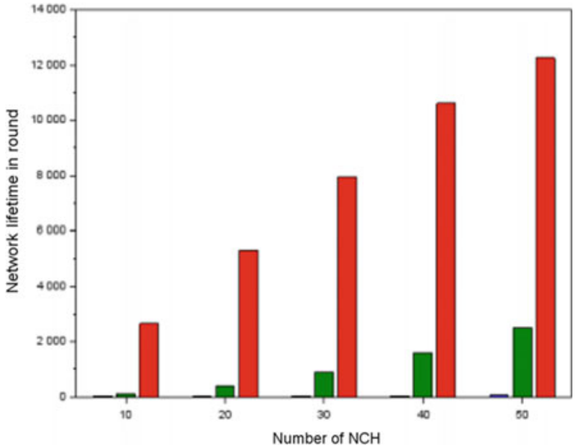


Fig. 12 Algorithm complexity in terms of number of comparisons

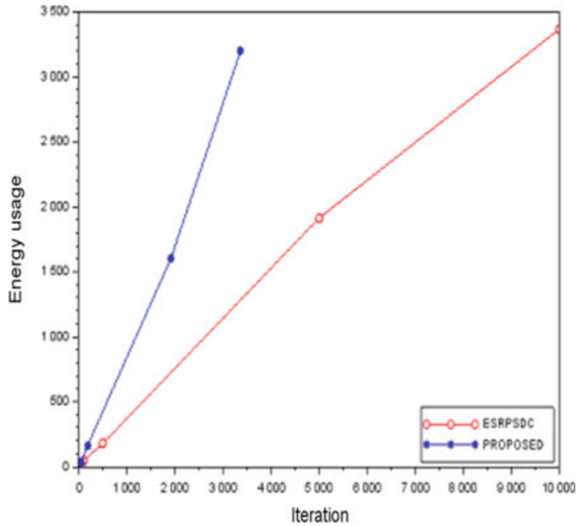


Fig. 13 Energy usage comparison

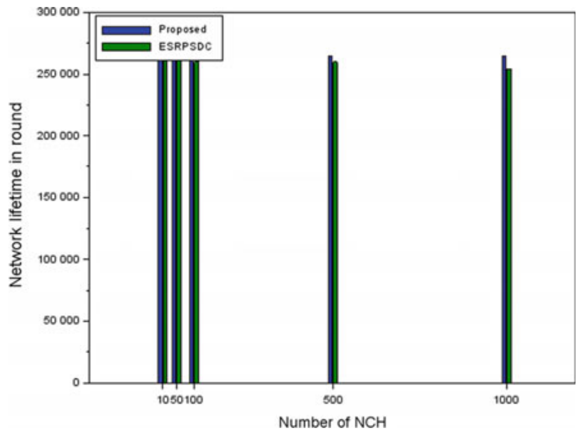


Fig. 14 Network lifetime comparison

References

1. Zhang, S., Zhang, H.: A review of wireless sensor networks and its applications. In: 2012 IEEE International Conference on Automation and Logistics, Aug 2012, pp. 386–389
2. Singh, S.P., Sharma, S.C.: A survey on cluster based routing protocols in wireless sensor networks. Procedia Comput. Sci. **45**, 687–695 (2015)
3. Azad, P., Sharma, V.: Cluster head selection in wireless sensor networks under fuzzy environment. ISRN Sens. Netw. (2013)
4. Jan, B., Farman, H., Javed, H., Montrucchio, B., Khan, M., Ali, S.: Energy efficient hierarchical clustering approaches in wireless sensor networks: a survey. Wirel. Commun. Mob. Comput. (2017)

5. Kanakaraju, R., Singh, A.V.: Clustering enhanced approach for network lifetime in wireless sensor network. In: 2017 IEEE International Conference on Circuits, Controls, and Communications (CCUBE), Dec 2017, pp. 195–199
6. Azharuddin, M., Kuila, P., Jana, P.K.: A distributed fault-tolerant clustering algorithm for wireless sensor networks. In: 2013 IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), Aug 2013, pp. 997–1002
7. Ganesh, S., Amutha, R.: Efficient and secure routing protocol for wireless sensor networks through SNR based dynamic clustering mechanisms. J. Commun. Netw. **15**(4), 422–429 (2013)
8. Sahu, B., Sahu, P., Dash, P.: Fault tolerant dynamic multi-hop clustering in under water sensor network. In: 2018 IEEE International Conference on Information Technology (ICIT), Dec 2018, pp. 101–105
9. Trinchero, D., Fiorelli, B., Galardini, A., Stefanelli, R.: Underground wireless sensor networks. In: 2009 IEEE 10th Annual Wireless and Microwave Technology Conference, Apr 2009, pp. 1–3
10. Karim, L., Nasser, N., Sheltami, T.: A fault-tolerant energy-efficient clustering protocol of a wireless sensor network. Wirel. Commun. Mob. Comput. **14**(2), 175–185 (2014)
11. Sahu, B., Pattnaik, R., Dash, P.: On-demand data transmission in multi-hop cluster in underwater sensor network. In: 2018 3rd IEEE International Conference for Convergence in Technology (I2CT), Apr 2018, pp. 1–5
12. Yadav, S., Kumar, V.: Optimal clustering in underwater wireless sensor networks: acoustic, EM and FSO communication compliant technique. IEEE Access **5**, 12761–12776 (2017)
13. Sahu, B., Khilar, P.M.: Selective data transmission in SNR based clustered-underwater wireless sensor network (CUWSN). In: 2015 IEEE International Conference on Man and Machine Interfacing (MAMI), Dec 2015, pp. 1–4
14. Garimella, R.M., Edla, D.R., Kuppili, V.: Energy Efficient Design of Wireless Sensor Network: Clustering. Centre for Communications International Institute of Information Technology Hyderabad, India (2018)
15. Ishaq, Z., Park, S., Yoo, Y.: A security framework for cluster-based wireless sensor networks against the selfishness problem. In: 2015 Seventh IEEE International Conference on Ubiquitous and Future Networks, July 2015, pp. 7–12
16. Varma, S., Tiwary, U.S.: Data aggregation in cluster based wireless sensor networks. In: Proceedings of the First International Conference on Intelligent Human Computer Interaction, pp. 391–400. Springer, New Delhi (2009)
17. Ganesh, S.R.A., Amutha, D.L.R.: Modified triple umpiring system for wireless sensor networks. PSG Tech Natl. J. Technol. **8**(1), 48–63 (2012)
18. Panda, B.S., Bhatta, B.K., Mishra, S.K.: Improved energy-efficient target coverage in wireless sensor networks. In: International Conference on Computational Science and Its Applications, July 2017, pp. 350–362. Springer, Cham (2017)
19. Kurniawan, A.: Selective route based on SNR with cross-layer scheme in wireless ad hoc network. J. Comput. Netw. Commun. (2017)
20. Swain, R.R., Mishra, S., Samal, T.K., Kabat, M.R.: An energy efficient advertisement based multichannel distributed MAC protocol for wireless sensor networks (Adv-MMAC). Wireless Pers. Commun. **95**(2), 655–682 (2017)

A Review on Mobile Cloud Computing

Napur Bharati, Sucheta Das, and Mahendra Kumar Gourisaria

Abstract Cloud means network or Internet. Cloud computing is defined as retrieving, accumulating, and handling the application online. It also offers online data storage, its foundation, and application. This provides a method by which the applications as profitable over the Internet are obtained. Mobile cloud computing is an ability which utilizes this cloud computing in mobile apps. Mobile cloud computing is a very essential trend these days. Mobile devices have plenty of drawbacks and limitations such as battery longevity, capacity to store, and bandwidth. This leads to the innovation of this model called MCC. It is a configuration in which both the data warehouse and handling occur outside the mobile gadget. It brings abundant computational means to mobile customers, network handler as well as cloud computing donors. This paper introduces an overview of mobile cloud and computing, its well-organized architectures, and basic applications.

Keywords Mobile cloud computing (MCC) · Mobile gadgets · Cloud computation · Mobile users · Safety

1 Introduction

Nowadays, an unbelievable growth has appeared in the evolution and expansion of mobile gadgets for instance laptops, cell phones, etc., as well as diversity of mobile and its computation, interlinking, and assured machinery. Mobile cloud and its computation is a subdivision of cloud computation that authorizes favorable, immediate network access and storehouse that can be further applied to provide

N. Bharati (✉) · S. Das · M. K. Gourisaria
School of Computer Engineering, KIIT Deemed to Be University, Bhubaneswar, India
e-mail: nupurmahanta47@gmail.com

S. Das
e-mail: suchetarumi09@gmail.com

M. K. Gourisaria
e-mail: mkgourisaria2010@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_21

distinct types of services and applications to the mobile customers [1]. Its aim is to supply the position-aware mobile facilities with assured mobile cloud assets, service operations, and data handling power-productive mobile cloud assets in a pay-as-you-use prototype. It is an integration of cloud computing and mobile domain presenting the smart phone users' new provisions and services on their gadgets that was tough to be attained in the absence of the existence of the cloud [2].

Unlike conventional mobile and its computation mechanisms, capitals in mobile cloud as well as its computing are supersaturated and allotted within a category of many circulated computing machinery instead of localized computing devices and responders. Along this, MCC has been acknowledged as the coming era's computing framework and also foundation.

To provide cloud facility in mobile environment, various issues and problems have to be confronted [3]. Equipment may hand off amid various wireless communication districts, and transfer carriers are not so authentic to assure cloud facility transportation [4]. Moreover, mobile gadgets cannot control complex implementations because of their instinctive character, and moreover it is impractical that mobile gadgets are every time networked, that is, the offline solutions for mobile devices are needed to be considered [5]. A few scheme applications partition and offload plots to leverage the job burden of cloud and user, which will minimize handling and dealing load on the users [6]. This paper instigates the fundamental prototype of mobile cloud and computing and studies [6] state of art of structures. Firstly, the need of MCC is outlined. Then, after launching the base structure of mobile cloud computing and its features, various challenges and modern days' applications are described (Fig. 1).

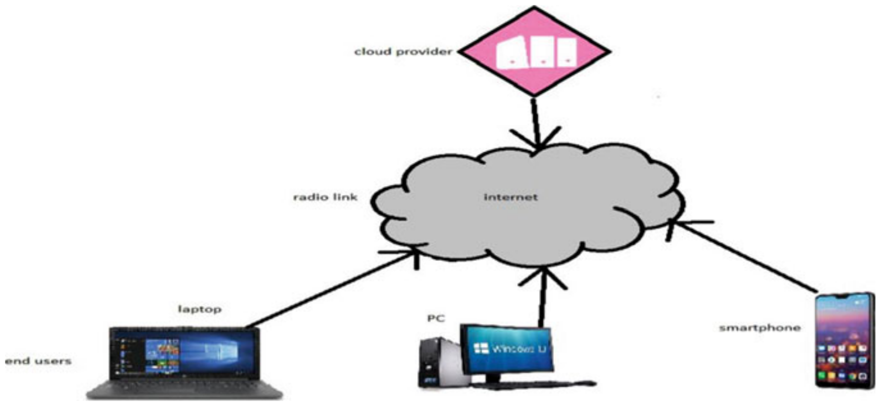


Fig. 1 Mobile cloud computing

2 Need of MCC

Nowadays, progress in cloud computing supplies remarkable advantages to mobile customers as cloud configuration and manifesto give effectively. Wide ranging computing potentiality [7], adaptable expandability and excessive resource splitting and uses proofs its efficiency. This will resolve a lot of usual and established drawbacks in mobile computing. Mobile cloud computing is in great need to meet application requirements in fields like image processing, sensor data application, multimedia hunt, and many more social networking platforms.

3 Architecture

The portable cloud and its processing is an amalgamation of mobile processing, cloud processing, and wireless Web technology. This combines the superiority of every above-cited mechanisms and thus be able to referred as cloud processing for mobiles gadgets [1]. The architecture basically comprises of: mobile computing, wireless technology in addition with cloud structure [7]. Data and information are being forwarded to the clouds via the wireless systems that are Internet and conversely while retrieving the data.

The mobile customer directs the data to the mobile gadgets. The data sent then moved to base transceiver (BTS) which are antenna or satellites and then passed to the central processor [8]. In the central processor, a complex mechanism is followed. Each data are given their data-ids, so that they are bifurcated from the other data which will resolve the problems during data segregation. Also the data timing and location of the user is being noted. Then, these are data that are transferred to the server where a replica of these is being created inside the database before sending them anywhere. [8] Mobile network controllers can give facilities to mobile customers via authentication, authorization, and accounting (AAA) under the home representative and followers statistics accumulated within database. Following this, the followers' or contributor's claims are being sent toward the cloud via the wireless system that is Internet. Afterward, appeals are being sent to the cloud for transferring data, and data itself are sent to the application server which serves as a connector in between cloud and wireless Web. And lastly, instructions and data are transferred to the clouds and accumulated in cloud data center (Fig. 2).

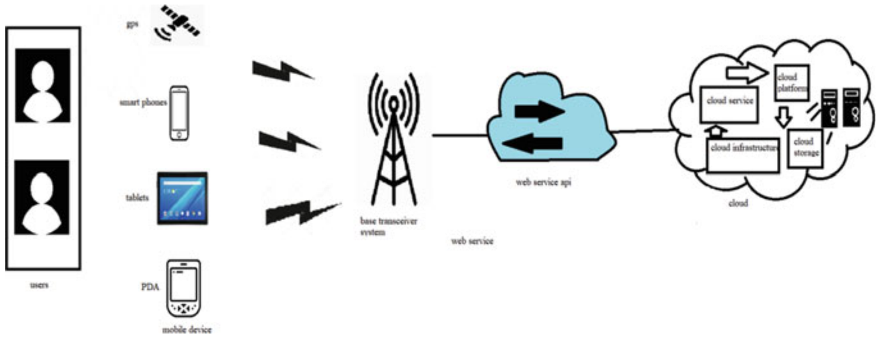


Fig. 2 MCC architecture

4 Background

4.1 Service Miniatures of MCC

Infrastructure as a service(Iaas): An ability supplied to the user is to arrangement's handling, [9] accumulation, networks, and further basic gauging measures where the user is enable to utilize and run random program that comprises of operating systems as well as implementations. Instances are Amazon Elastic Compute, Rack Space, and IBM computing on claim.

Platform as a service (Paas): An ability supplied toward the customer is to utilize the cloud foundation user-generated and obtained implementations made with the help of programming functions and tools assisted by the supplier. Pass contributes progress stage which organizes fully and in advanced cloud implementations. Instance is Google App Engine [9].

Software as a service (SaaS): An ability supplied toward the user is to utilize the contributor's implementations employed over a cloud foundation. The users do not supervise or handle the fundamental cloud foundation accompanied by feasible anomaly of finite customer-precise implementation settings. Example is Microsoft office 365.

4.2 Deployment Miniatures of MCC

Private Cloud: Cloud foundation runs uniquely for an administration. This is built peculiarly for an administration inside its private information station [9]. The organizations control all the cloud assets that they posses.

Public cloud: The cloud foundation is created accessible for the normal people or extended production collaboration [10] and is possessed by an association vending cloud facilities.

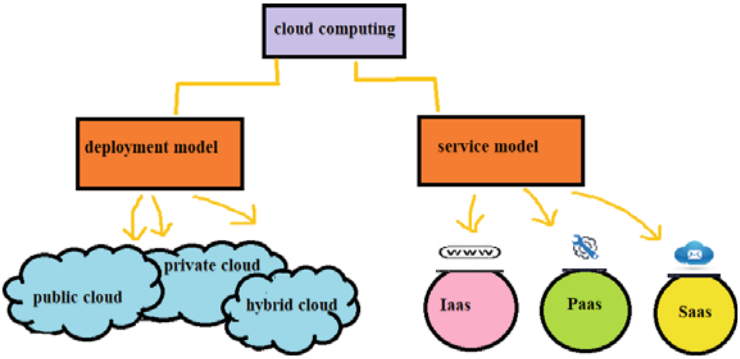


Fig. 3 Cloud service and deployment miniatures

Community cloud: The cloud framework [11] is splitted by various groups and helps a certain section that has sharing and divided interests and regards. The cloud association establishes within an extent of profitable scalability and autonomous stability.

Hybrid cloud: Cloud foundation is just a mixture of more than two clouds that stands as distinctive systems yet is heaped jointly by normal and systematized technologies that allows facts, data, and application transportability (Fig. 3).

5 Features of Mobile Cloud and Its Computing

5.1 Scalability

Mobile applications can be climbed up and climbed down to encounter unforeseeable user demands. Scalability comprises of three aspects: Cloud scalability, Web scalability, and portable scalability in words of mobile customers as well as gadgets. Since due to the immensely scalable feature of MCC, numerous corporations are currently depending on handled data and information centers where they are cloud specialists trend in continuing and scaling splitted, private as well as hybrid clouds [11].

5.2 Mobility

Mobile and portable junctions in mobile processing systems may initiate interrelation with rests even with attached and fixed and predetermined nodes in cabled system via mobile assist sites throughout their working [12]. Mobile clouds allow mobile customers for retrieving mobile cloud applications as well as facilities no matter when or where with personalized approachability.

5.3 Mobile Cyber Security with Privacy

It mentions the major structure of security potentials, automation, and procedures and operations planned for securing mobile gadgets, varied networks, cloud servers, and data from attack, destruction, and unapproved access [10]. Mobile cloud computing security concerns can be classified as two distinct groups: mobile user protection and securing and data on cloud servers.

5.4 Enhancing Reliability with Mobile Computing

By accumulating data or managing applications on clouds, great reliability is retained since data and applications are piled and are aided on several computers, and this will diminish the possibility of data loss [13]. The cloud computing supplies a secure manner to accumulate customer's data while the customers do not panic regarding the matters like software and program renovating, leak patching and virus attack, and reduction of data [14].

5.5 Reduced Cost

Mobile cloud and computing equally assists in lessening the working fare for worked out thorough applications which takes [15] prolonged schedule and a big sum power when executed over the finite-resource gadgets.

5.6 Multiple Occupancy

Multiple occupancy property permits solitary mobile cloud program exemplar to work for several mobile occupants on a wireless internet or diverse network [16]. Service contributors can split the accumulated resources and values to assist a diversity of implementation and applications and huge total of customers.

5.7 *Imbalance Web Transmission*

Servers and acquiring nodes and further mobile support stations permit a well-built dispatch/receive potentiality, while alike potentiality in mobile points is fairly frail approximately [16]. Therefore, the transmission rate of data as well as overhead connecting downlink and uplink is inconsistent.

6 Why Mobile Cloud Computing?

Mobile cloud and computing assists beating drawbacks of mobile gadgets especially of the handling energy and instruction accumulation [15]. Several of the latest technical and applied features could be supplied by portable clouds. Specifically, arranging of circumstances and situations as well as position consciousness allows customization of services which is a fascinating feature. Trading can be further strengthful as well as highly organized by assimilating mobile cloud computation in it [16]. The motivation of MCC is to focus the requirements in raising the handling power and battery existence span of mobile gadgets. The Splitting of mobile data and implementations, cloud applications, and SAAS structures by associating to multiple detector webs as well as mobile gadgets such that mobile allowed active and quick instructions applications can be effortlessly evolved and placed in several mobile applications.

7 Issues in MCC

7.1 *Scarcity of Mobile Appliances*

While considering mobile gadgets together with clouds, the first and foremost thing is the limitations and restrain of resources. [18] Mostly mobile appliances have restrictions with regard to processing potential and capability, accumulation, restricted battery capability, mediocre visibility in contrast to computer.

7.2 Safety and Seclusion

Every webbing action is vulnerable to one or more kinds of harmful and spiteful attacks [18]. It is becoming hard and toilsome to handle threats on mobile gadgets in contrast to desktop gadgets [19] since in a Wi-Fi network the probability of nonexistence of data from the network is more.

7.3 Short Data Transferral Rate (Bandwidth)

In a transmission Web and network, data transfer amount is a crucial aspect [20]. Since there is a restriction of transferral rate, distribution of this minimal bandwidth amid the numerous mobile users and consumers sited in the similar regions or workplaces and certainly engaged in a similar content is to be shifted, which guarantees in the betterment of peculiarity, and hence, this explanation is implemented predominantly for the circumstances where users within a particular zone are concerned into the similar content [21].

7.4 Offloading

This is one of the significant specificity of MCC results in rising of battery operating time and enhances the execution of the applications [17]. Yet, this idea cannot be viewed as an efficient way to economize the power and raise mobile gadgets implementation as seen in previous surveys. There is no access to mobile customers over the unloading and offloading operations resulting in rising of hazards of illegal control to unloaded content.

7.5 Quality of Communication and Broadcasting

As compared to wired and cabled network which employs physical and real links to guarantees uniformity of data transferral rate, the bandwidth in MCC surroundings is repeatedly altering as well as the links are irregular since there is presence of clearance in network detain [21]. Latency and climate state cause variations in data transmission quantity and delay in network.

8 Conclusion

Mobile cloud computing is classified among the prominent, revolutionary, and pioneering technology that is an amalgam miniature of cloud and mobile computing. This has emerged as a trendy research subject these days since there is an extreme utilization of mobile appliances by a huge number of customers [19]. The expanding utilization of computer network and accumulation of data moving to cloud environs contributes to the enormous evolution of MCC [8]. It works for the motive of granting ample portable computing via merge transmission betwixt front and end customers despite of varied, Wi-Fi surroundings and basic platforms in worldwide roaming. This paper comprises of basic outline of MCC, its necessity, fundamental architecture, its framework [12] which includes service and deployment miniature, characteristics, and key issues associated with it. Mobile cloud computing is interpreted to be a potential and modern mechanism in upcoming days.

References

1. Sharma, P.S., Bothra, S.U.: Mobile cloud computing: its challenges and solutions. Int. J. Comput. Sci. Mob. Comput. **4**(5), 287–293 (2015)
2. Prasad, M.R., Gyani, J., Murti, P.R.K.: Mobile cloud computing: implications and challenges. J. Inf. Eng. Appl. **2**(7), 7–15 (2012)
3. Le, G., Ke, X., Song, M., Song, J.: A survey of research on mobile cloud computing. In: 2011 10th IEEE/ACIS International Conference on Computer and Information Science (2011)
4. Satyanarayanan, M.: Mobile computing: the next decade. In: Proceedings of the 1st ACM Workshop on Mobile Cloud Computing and Services: Social Networks and Beyond (MCS) (2010)
5. Forman, G.H., Zahorjan, J.: The challenges of mobile computing. IEEE Comput. Soc. Mag. **27**(4), 38–47 (1994)
6. Al-Janabi, S., Al-Shourbaji, I., Shojafar, M., Abdelhag, M.: Mobile cloud computing: challenges and future research directions. In: 2017 10th International Conference on Developments in eSystems Engineering (DeSE) (2017)
7. Chang, R.S., Gao, J., Gruhn, V., He, J., Roussa, G., Tsai, W.T.: Mobile cloud computing research issues, challenges and needs. In: 2013 IEEE Seventh International Symposium on Service-Oriented System Engineering (2013)
8. Mubaik, A., Hasibuan, Z.A.: Government services integration based on cloud technology. In: Proceedings of the 14th International Conference on Information Integration and Web-Based Applications and Services-IIWAS 12 (2012)
9. Qureshi, S.S., Ahmad, T., Rafique, K., Aslam, S.U.: Mobile cloud computing as future for mobile applications—implementation methods and challenging issues. In: Proceedings of IEEE CCISS 2011
10. Qi, H., Gani, A.: Research on mobile cloud computing: review. Trend Perspect
11. Zhang, W., Chen, Q.: From E-government to C-government vi cloud computing. In: 2010 International Conference on E-Business and E-Government (2010)
12. Huand, D.: Mobile Cloud Computing. IEEE COMSOC MMTC ELetter. Available. http://www.comsoc.org/~mmc/
13. Sharma, M., Bhopi, P.: Study on mobile cloud computing, it's architecture, challenges and various trends. Int. Res. J. Eng. Technol. **04**(06) (2017)

14. Li, S., Gao, J.: Moving from mobile databases to mobile cloud data services. In: 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (2015)
15. Chang, R.S., Gao, J., Gruhn, V., He, J., Roussos, G., Tsai, W.T.: Mobile cloud computing research—issues, challenges and needs. In: 2013 IEEE 7th International Symposium on Service-Oriented System Engineering (2013)
16. Srivastava, P., Khan, R.: A review paper on cloud computing. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **8**(6) (2018)
17. Satyanarayanan, M.: Fundamental challenges in mobile computing. In: Proceedings of the 5th annual ACM symposium on principles of distributed computing, pp. 1–7 (1996)
18. Springer Science and Business Media LLC 2019. Computational Intelligence in Data Mining
19. Gupta, P., Gupta, S.: Mobile cloud computing: the future of cloud. Int. J. Adv. Res. Electr. Electron. Instrum. Eng. **1**(3), 134–145 (2012)
20. El-Sofany, H.F., El-Seoud, S.A.: Studying the effectiveness of using mobile cloud computing applications in learning. In: 2016 International Conference on Interactive Mobile Communication, Technologies and Learning (IMCL) (2016)
21. Mishra, S., Mohapatra, S.K., Mishra, B.K., Sahoo, S.: Chapter 5 Analysis of mobile cloud computing. IGI Global (2018)

Corroborating the Veracity of Body Fat Percentage and Classification of Sarcopenia Using Island Differential Evolution Algorithm

J. Grace Hannah and D. Gladis

Abstract Body fat is a decisive determinant in the identification of umpteen health ailments. A euphemism copiously addressed in various parts of the world with colossal awareness for agglomeration of body fat in the human body is obesity. It can be termed as a health gremlin that originates from physical stasis, detrimental food habits, genetic constitution and retrenches an individual's diurnal schedule. For a person to remain healthy, continual observation and analysis of body fat are incumbent. The quondam work of study exhorted the use of parameters such as BMI, age and gender to obtain the body fat percentage for an individual. But the deficit of this existing formula in discerning a healthy person with more muscles and less body fat, from an obese person is a momentous impediment to scientific progress and well-being. This paper asseverates the analysis of residuals and goodness-of-fit for the existing body fat formula to that of the parameter optimized derived body fat formula. Thereby substantiating that the melioration of the variables using a genetic algorithm corresponds to the impeccable standards established by using lipid counts from the body and the Bio-Electric Impedance Analysis (BIA) method. The above exploration has also aided in dissecting a vital integrant from the BIA method known as the muscle mass percentage which has envisaged the recognition of a disease called sarcopenia. The proposed study also stratifies sarcopenia in adult patients and produces the classification performance using the Island Differential Evolution (IDE) algorithm. Blood samples for lipid tests and the values from the BIA method from adult patients were drawn from The Institute of Biochemistry, Madras Medical College. The simulation results are efficiently acquired from MATLAB GUI.

Keywords Obesity · Genetic algorithm · Residuals and goodness-of-fit · Sarcopenia · Island differential evolution

J. Grace Hannah (✉)
Presidency College, University of Madras, Chennai, India
e-mail: hannahgrace161289@gmail.com

D. Gladis
Bharathi Women's College, Chennai, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_22

1 Introduction

Bariatrics is the science of dealing with the concomitant surgical procedures of obesity [1, 2]. A survey by the World Health Organization (WHO) depicted that 41 million children under the age of five were obese or overweight [3, 4] and that the rate at which obesity affects human beings is colossal to that of infirmities caused due to underweight. The bariatric medicaments utilized in recent years for mitigating these disorders have evolved multifariously. However, the phase of life where an individual tends to put on weight is that measure which would determine their potential to metamorphose back to normal. The calorie disparity observed between the ingestion, and the energy expended by an individual has engendered the amassing of adipose tissues in the human body.

The World Health Organization also declared that a whooping number of about 1.9 billion adults are overweight, 650 million are considered to be obese [4, 5] and about 80% of them have type 2 diabetes and other associated ailments. An individual having a higher waist-hip ratio and weighing more than the reckoned standard of measure usually finds it laboriously exhausting to lose weight. The cognizance of obesity is homologous to the assimilation of the knowledge of adipose tissues. Obesity is categorized into two: Hyperplasic and Hypertrophic [6, 7], where the former is the inflated, unbroken energy which is disintegrated and converted into new cells, while the latter denotes the uncurbed, proliferated fat. The hypertrophic obesity comprehends the deciphering of adipose tissues which are multifaceted and form a complex organ [4, 8, 9] bisecting into White Adipose Tissues (WAT) and Brown Adipose Tissues (BAT) (Fig. 1).

Besides operating as a system for caching the nimiety of energy, defense against frigid conditions and the health perils confronting our day to day lives, adipose tissues are responsible to generate a conglomeration of molecular emblematic called as adipokines. These adipokines steer diversified medley of functions comprising of appetite, fertility, neural development, inflammatory responses and the action of other hormones including insulin [9, 10]. Despite their lucrative functions, the surplus energy amassed and the White Adipose Tissues exacerbates the various health detriments [11, 12]. Although multifarious studies depict that the fat cells are waned only in proportion, obliterating the fat cells which are accumulated becomes an increasingly onerous task [2, 13]. Due to these predicaments, the analysis of fat percentage in the human body is rudimentary to evince the health fettle of an individual [14, 15]. Figure 2 depicts the classification of fat deposition in the human body [2, 4, 16–18].

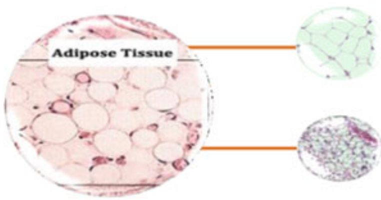


Fig. 1 Classification of adipocyte tissues

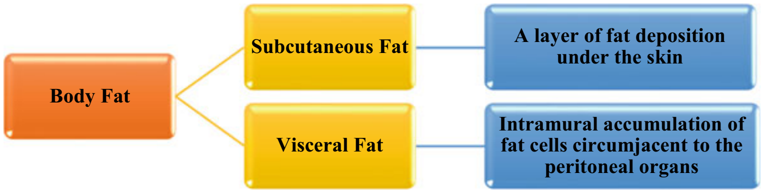


Fig. 2 Classification of body fat based on accumulation in the human body

Although multifarious techniques for prophylactic measures on obesity are subsumed, this paper aims to demonstrate through residual calculation and goodness-of-fit that the novel formula derived for body fat meets the standard scales of accuracy after the optimization of parameters. The identification of sarcopenia from the optimized formula and its classification is explicitly delineated through the IDE algorithm. Thus, enabling to fathom the occurrence of sarcopenia in individuals having higher body fat percentage and establishing the accuracy of classification using the optimized novel formula.

The sequence of the paper is as follows: Section 2 briefs the methodologies used for sample collection and catalogs the components obtained from the BIA method. Section 3 elucidates the techniques used for variable optimization, obtaining the correlation coefficient along with the curve fitting model, and the analysis of agnizing sarcopenia using the IDE algorithm. Section 4 demonstrates the results obtained in MATLAB, and Sect. 5 interprets the compendium of results and conclusion of the work done.

2 Methodology

The samples of blood tests and factors from the BIA method were drawn from adults aged between 20 and 75 years. The population of study was circumscribed to obese, hyperlipidemic patients who were suffering from health gremlins ascribed to an agglomeration of fat in the body. This study enumerates the formulation of body fat percentage by computation of the values procured from the blood tests and the Bio-Electric Impedance Analysis (BIA) method for an individual (Fig. 3).

The above two methodologies are ostracized for individuals who were anemic and suffering from other pernicious health disorders, women who are pregnant, and patients with hemorrhage dysfunctions. The lipid profile tests and the BIA method catalogs a list of components for each individual which helps in analyzing the state of health of an individual [4, 19]. Both the above tests also aids to sift the distribution of fat cells in the body and delineates the following factors: Total Cholesterol (TC), High-density Lipoprotein (HDL), Low-density Lipoprotein (LDL) and Triglycerides (TG) are fragmented integrants of the former, whereas the latter tests generate values by passing a small percentage of electric current through the body [2, 4, 20–24]. The

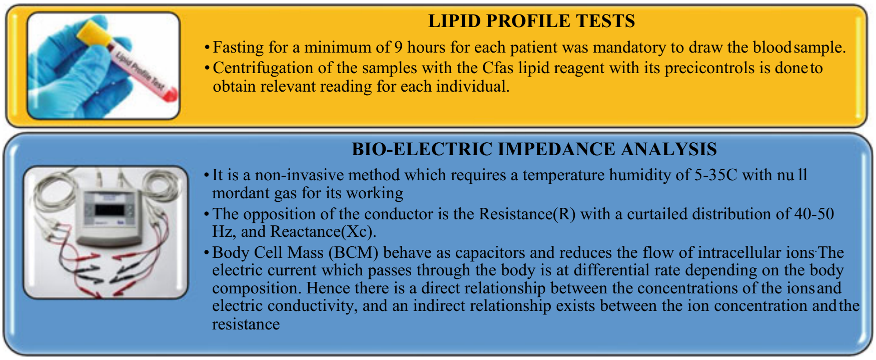


Fig. 3 Methods used for collection of data samples

values for elements such as subcutaneous fat, visceral fat percentage, the concomitant age corresponding to the fat percentage, Body Mass Index (BMI) and the muscle mass percentage for every individual are obtained from the BIA method.

3 Optimization of the Formula and Techniques Used

The existing body fat formula is given as:

$$\left[\left(1.2^{*}\mathrm{BMI}\right) + \left(0.23^{*}\mathrm{Age}\right) - \left(10.8^{*}\mathrm{Gender}\right) - 5.4\right] \tag{1}$$

The conventional formula derived by Paul Duerenberg [4, 24, 25] has been permeated to anatomize the fat percentage in the human body. But it has been ascertained that for a person of herculean build with more muscle mass than fat cells, there arouses an unwarranted surge in the BMI, ensuring that the individual falls in the anamously obese or overweight division. Therefore, to circumvent errors and to proffer an unswerving precise result, a novel formula comprising of Total Cholesterol (TC) from the lipid profile tests, subcutaneous and visceral fat from the BIA method, age pertaining to the fat accumulation in the body and gender of the person are taken into consideration [2, 4, 24].

$$\begin{aligned} \text{Bodyfat}\ \% = {} & [0.5(\mathrm{SF} + \mathrm{VF}) + (0.03(\mathrm{AGE(M)} \\ & - \mathrm{AGE})) + (0.02 * \mathrm{TC}) - (10.9 * \mathrm{GEN}) + 11.8] \end{aligned} \tag{2}$$

where SF and VF signify the subcutaneous and visceral fat, age (M) is the age of the person with respect to the body fat percentage, gender (Gen) takes the value of 0 for female and 1 for male [2, 4, 24].

4 Regression and Genetic Algorithm

Variable optimization is an integral part of meliorating the coherent body fat formula. Therefore, in order to correlate the derived formula and to obtain faultless results, the curve fitting model along with genetic algorithm has been implemented. Optimization using genetic algorithm has been executed for selecting the strong chromosomes by using genetic operators through the Mean Square Error (MSE) fitness function. The chromosomes are then entwined by implementing the arithmetic cross over method, with the cross over probability set to 0.8. The final step of genetic operation is to use mutation to randomly commute the genes from the selected pair of chromosomes. The odds of processing the mutation is earmarked to 0.2 and the generation of iterations is set to 1000, with the elite count set to 4. Optimization occurs with the number of generations and the best fitness value and the mean fitness value are obtained. Least—Square regression method is applied to fit the curve between the obtained and the forecasted values.

Initially, the predicted values are found by inversing the original data matrix and by applying the multiplication of the matrix with the unknown parameters. Then the mean square error is found by squaring the variance between the original data and the forecasted values.

5 Residuals and Goodness-of-Fit

Once the optimization of the variables is done, the post-processing of the derived, optimized parameters needs to be checked. And hence the residuals play a vital role. Residuals are the difference between the observed values of the response or dependent variable and the value that a model predicts. A least-square fit is obtained by reducing the sum of squares of the residuals. One of the measures used is the Goodness-of-Fit (GoF) [26] which defines the coefficient of determination and is called R^2. GoF determines and indicates the closeness of the obtained values from fitting a model to that of the dependent variable the model intends to predict [26].

$$R^2 = 1 - \mathrm{SS}_{\mathrm{resid}}/\mathrm{SS}_{\mathrm{total}} \tag{3}$$

where $\mathrm{SS}_{\mathrm{resid}}$ is the sum of the squared residuals from the regression and $\mathrm{SS}_{\mathrm{total}}$ is the sum of the squared differences from the mean of the dependent variable. After the estimation of parameters, the goodness-of-fit checks to diagnose the adequacy of the fitted model and aids in giving further suggestions to augment the fitness of the fitted model [26].

6 Identification and Classification of Sarcopenia

Sarcopenia is a term that refers to the degeneration of muscles [27]. A salient component obtained from the BIA method is the muscle mass percentage for each individual. It has been analyzed that after the parameter optimization, there is a significant progression in terms of accuracy with the body fat percentage. The analysis evinced that the values for muscle mass percentage taper off with the increase of body fat percentage for an individual. Thus, indicating that the loss of muscles is proportionally related to the accretion of fat cells in the human body. The proposed study stratifies sarcopenia as 'Positive' indicating that there is muscle atrophy due to intensified accruement of adipose tissues, and 'Normal' signifying that muscle deterioration is not seen. The classification accuracy is computed using the island differential evolution in a feed-forward neural network. It has been studied that when the formula has been contoured using optimization algorithms, the values for muscle mass percentage amortizes with the escalation of body fat percentage.

Tables 1 and 2 show the muscle mass percentage with relation to the body fat percentage of an individual garnered from the Bio-Electric Impedance Analysis. The tables patently expound that with the existing formula in Table 1, the recognition and extant of sarcopenia are misclassified due to the absence of tangible attributes and variable optimization.

Table 1 Analysis of sarcopenia using the existing body fat formula

Gen	Age	Existing formula	Muscle mass	Sarcopenia analysis
1	25	24.58	27.60	Normal
0	49	43.33	21.20	Positive
1	54	33.54	25.20	Normal
1	60	29.16	24.32	Normal
0	52	40.52	23.32	Positive
0	54	24.58	35.72	Normal

Table 2 Analysis of sarcopenia using the derived body fat formula

Gen	Age	Derived formula	Muscle mass	Sarcopenia analysis
1	25	30.41	20.20	Positive
0	49	54.09	16.60	Positive
1	54	40.98	21.40	Positive
1	60	24.73	26.50	Normal
0	52	46.40	19.40	Positive
0	54	44.51	22.70	Positive

7 Island Differential Evolution Algorithm

The island differential evolution (IDE) algorithm is an accretion of genetic algorithms with the appendage of migrations introduced to sub-populations. Migrations between islands aids in the fraternizing of individuals and disseminates information about the superlative individuals on each island. This approach of transhumance commingled with the features of Differential Evolution (DE) yields new offspring by crossover, and weeds out poor solutions. The parameters of an island model are as follows:

Size of Transhumance: An epoch-making facet of the IDE algorithm is the measure of sub-population migrations between islands. The size of transhumance plays a profound assay to the classification efficacy. If the migration of population is paltry, then the ascendancy of migration is trifling, but if the exchange of sub-population is very high, then the global diversity which is the prominent aspect of the IDE algorithm debilitates [28, 29]. The proposed study probes into the appropriate size of migration to obtain unerring results.

Migration Strategy: In order to actuate the migration between islands, the individuals participating in the transhumance must be selected. The next step involves the migration strategy for the participating individuals to abide in an island. There are a few strategies for replacement of individuals in islands [28, 29]. This study focuses on selecting the best individuals and replacing them with the worst.

Topology of Migration: This paper pivots on the ring topology, among the other migration topologies. The migration topologies such as random, torus and fully connected are implemented in various disciplines [28, 29].

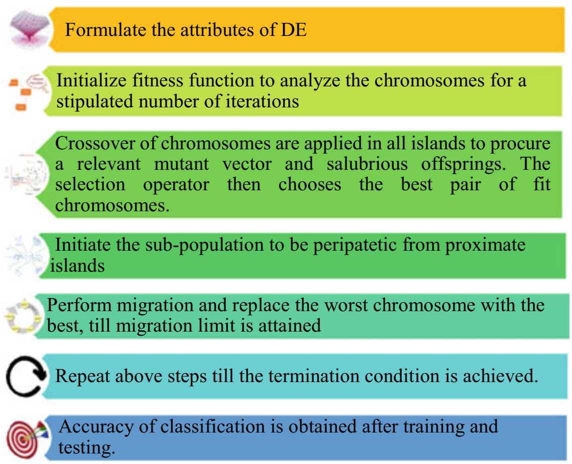
Transhumance Interlude: The method of migration can either be done in an asynchronous or in a synchronous manner, which depicts that after every ith iteration a migration can take place or at any juncture, respectively. It is often conceded that recurrent migrations steer the way to swift convergence and higher selection. But the cons accompanied by faster convergence and higher selection of individuals is the propensity of getting trapped in local optima [28, 29]. This paper cynosures on the synchronous interval where migrations take place after every ith iteration (Fig. 4).

The above figure illustrates the working of the IDE algorithm. The stopping criteria mentioned for the termination of the algorithm works on the below two facets:

- Convergence criteria that state the error between two previous generations should be less than the specified value
- Upper bound on the number of generations

The initial parameters of DE are training size percentage, the number of neurons, and the number of iterations. The objective fitness function used is the Mean Square Error (MSE) performance method. The crossover operator is applied to obtain new individuals. The crossover acceptance rate takes the value 0.1 for better results. The following snippet shows the offspring acceptance after crossover.

Fig. 4 Procedure of approach using island differential evolution (IDE) model




```
GE = pop{i}(ri,j) + F * (pop{i}(r(2),j) - pop{i}(r(3),j));
if (rand < CR)
cs(j) = GE;
else
cs(j) = pop{i}(ri,j);
```

where GE signifies the offspring obtained after crossover, CR represents the crossover constant and the pop$\{i\}(r_{i,j})$ symbolizes the parent chromosome. The chromosome value is adjusted by adding and subtracting with another individual from the population. If the chromosome 'GE' is better than the parent chromosome then it is replaced and taken as the offspring, but instead, if the parent chromosome proves to be better than the offspring GE obtained, then the parent is taken into consideration for further processing.

The next step involves the migration of the individuals between islands. The number of islands taken for the study spanned from two to ten and has been amended frequently so as to find the optimum value. The selection probability indicates the quantum of the selection of chromosomes from each iteration. The above inputs are then trained using a feed-forward neural network and the classification accuracy is procured for the training and testing phases (Table 3).

The above table is a succinct tabulation of the best performances obtained using the IDE algorithm through a feed-forward neural network. The values have been permuted and used on the population size (pop. Size), number of islands and selection probability, thereby acquiring the results for fitness and classification performances.

Table 3 Values obtained after testing and training of feed-forward neural network using IDE

S. No.	Pop. Size	No. of islands	Migration %	CR	Selection probability	Fitness value	Training accuracy	Training PPV	Testing accuracy	Testing PPV
1	24	10	10	0.1	0.7	0.1345	0.8400	0.9512	0.9467	0.9963
2	22	8	10	0.1	0.6	0.1410	0.8229	0.9274	0.8267	0.9074
3	15	7	10	0.1	0.6	0.1415	0.8114	0.9194	0.8133	0.8959
4	14	6	10	0.1	0.6	0.1434	0.8107	0.9089	0.8800	0.8830
5	13	4	10	0.1	0.5	0.1428	0.8043	0.8912	0.8133	0.8148

8 Results Obtained

The above GUI Fig. 5 shows anatomization of the parameter optimization. The genetic algorithm parameters such as number of variables, initial population, crossover method, mutation and cross over probabilities along with the elite count and the dataset are given as input to the optimization algorithm. The output yields the best fitness value and the meliorated values for the parameters of the body fat formula.

Figure 6 shows the post-processing step such as the computation of residual and goodness-of-fit (GoF) analysis. The result demonstrates the correlation coefficient along with the mean square error indicating that the derived model is impeccable to be equated with the standard caliber.

Figure 7 depicts the plotting of the values after the residual and goodness-of-fit analysis. This illustrates the values observed between the existing body fat formula using BMI to that of the optimized, unconventional novel body fat formula using lipids and BIA method.

Figure 8 illustrates the process of neural networks with the seven input attributes feeding the value as input to its subsequent hidden layer, and the values obtained from the hidden layer are passed on to the output layer, thereby creating a mapping

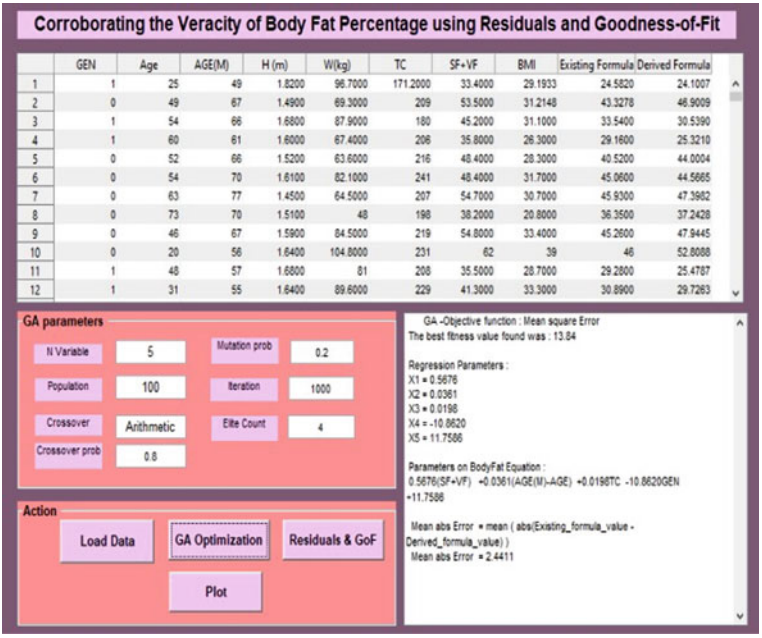
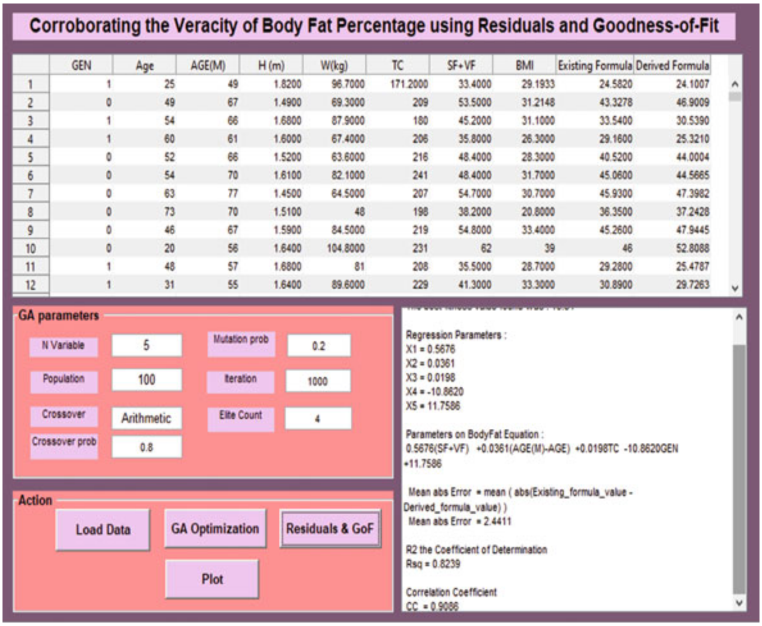


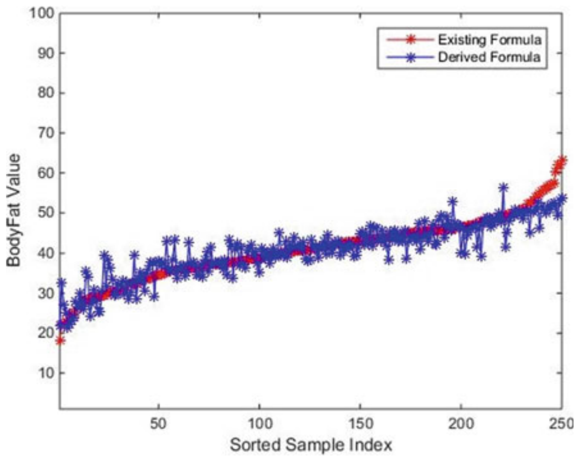
Fig. 5 Variable optimization using genetic algorithm



	GEN	Age	AGE(M)	H (m)	W(kg)	TC	SF+VF	BMI	Existing Formula	Derived Formula
1	1	25	49	1.8200	96.7000	171.2000	33.4000	29.1933	24.5820	24.1007
2	0	49	67	1.4900	69.3000	209	53.5000	31.2148	43.3278	46.9009
3	1	54	66	1.6800	87.9000	180	45.2000	31.1000	33.5400	30.5390
4	1	60	61	1.6000	67.4000	206	35.8000	26.3000	29.1600	25.3210
5	0	52	66	1.5200	63.6000	216	48.4000	28.3000	40.5200	44.0004
6	0	54	70	1.6100	82.1000	241	48.4000	31.7000	45.0600	44.5665
7	0	63	77	1.4500	64.5000	207	54.7000	30.7000	45.9300	47.3982
8	0	73	70	1.5100	48	198	38.2000	20.8000	36.3500	37.2428
9	0	46	67	1.5900	84.5000	219	54.8000	33.4000	45.2600	47.9445
10	0	20	56	1.6400	104.8000	231	62	39	46	52.8088
11	1	48	57	1.6800	81	208	35.5000	28.7000	29.2800	25.4787
12	1	31	55	1.6400	89.6000	229	41.3000	33.3000	30.8900	29.7263

Fig. 6 Values obtained from residuals and goodness-of-fit

Fig. 7 Plotting of existing formula versus derived formula



function between the input and the output layers. The parameters of the feed-forward neural network are the number of hidden neurons, training size and the number of iterations.

Figure 9 depicts the island evolution algorithm with the arguments of the feed-forward neural network and the parameters of the IDE algorithm given to procure the classification performance of sarcopenia in individuals.

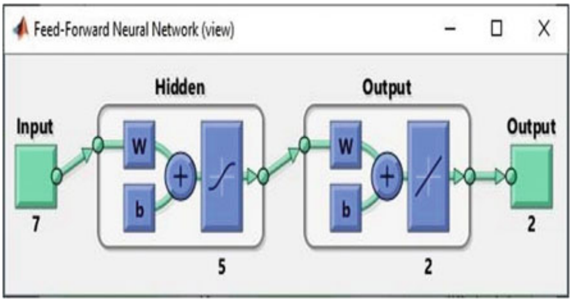


Fig. 8 IDE training feed-forward network

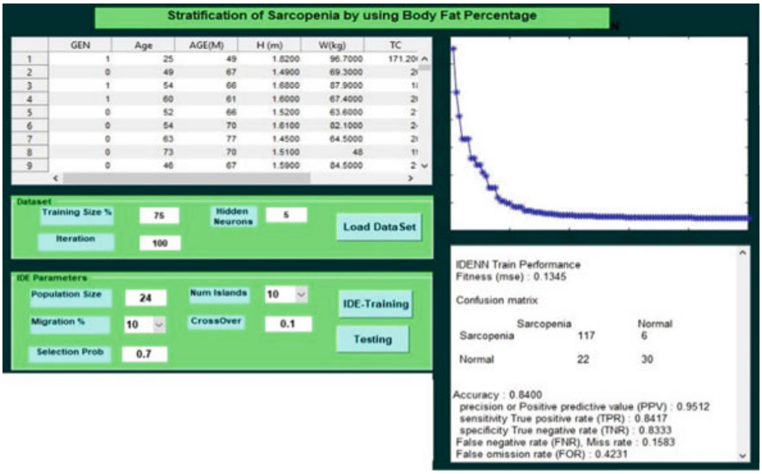


Fig. 9 Island differential evolution training and classification with best fitness value obtained using mean square error method

9 Conclusion

The analysis of this study has enlightened that when blood lipids attenuate, the body fat percentage escalates, thereby exacerbating the muscle mass percentage of an individual. This gives rise to sarcopenia in an individual. The precision of the body fat percentage is further augmented by optimization of the variables using regression and genetic algorithm. The estimation of residuals and goodness-of-fit has produced the variance and correlation coefficient of the previously established model to that of the derived model. The mean absolute difference of less than 0.7% between the existing body fat formula and the derived body fat formula has demonstrated that the novel formula after variable optimization and post-processing steps is at par with the standard measures. The novel body fat percentage has been surveyed by

Laparoscopic surgeons in Chennai and is put to test and use for various evaluations of analyzing body fat and its concomitant health gremlins in hospitals.

The IDE algorithm has facilitated the classification of sarcopenia effectively. The indagation itself has revamped the process of stratification by migration of sub-population to obtain global diversity within islands. The proposed study has also certitude that the optimized parameters of body fat percentage using the derived novel formula have helped further in the accurate classification of sarcopenia in individuals. The IDE algorithm has also corroborated through observation that larger number of islands emanate inimitable fitness values, increased classification accuracy and error rate as low as 0.3%. The subsequent work of study is to anatomize the sarcopenia dataset to juxtapose with other classification algorithms, and perceive the inadequacy in reaching the zenith of classification performance, and to obtain more cognizance regarding the relationship between lipids and muscles.

References

1. Definition and facts for bariatric surgery. https://www.niddk.nih.gov/health-information/weight-management/bariatric-surgery/definition-facts
2. Hannah, J.G., Gladis, D.: A statistical indagation of body fat percentage: a sift cognitive correlation of lipid data and bio-electric impedance analysis in humans. Int. J. Adv. Sci. Eng. Technol. (2018)
3. Obesity and Overweight. http://www.who.int/mediacentre/factsheets/fs311/en/
4. Hannah, J.G., Gladis, D.: Aggrandizing the accuracy of body fat percentage by stratification using decision tree. Int. J. Innov. Technol. Expl. Eng. (IJITEE) **8**(11), 1883–1886 (2019). ISSN: 2278-3075
5. https://www.who.int/factsonobesity
6. Drolet, R., Richard, C., Sniderman, A.D., et al.: Hypertrophy and hyperplasia of abdominal adipose tissues in women. Int. J. Obe. **32**, 283–291. https://doi.org/10.1038/sj.ijo.0803708. Published online 28 Aug 2007 (2008)
7. Mendes-dos-Santos, C.T., et al.: Normalization of height and excess body fat in children with salt-wasting 21-hydroxylase deficiency. J. Pediatr. (Rio J.) **87**(3) (2011). (Porto Alegre May/June 2011)
8. Cinti, S.: The adipose organ at a glance. Dis. Model Mech. **5**, 588–594 (2012)
9. Thomas, E.L., Fitzpatrick, J.A., Malik, S.J., Taylor-Robinson, S.D., Bell, J.D.: Whole body fat: content and distribution. Prog. Nucl. Magn. Reson. Spectrosc. **73**, 56–80 (2013)
10. Ouchi, N., et al.: Adipokines in inflammation and metabolic disease. Nat. Rev. Immunol. **11**, 85–97 (2011)
11. Bernlohr, D.A., et al.: Adipose Tissue and Lipid Metabolism. Elsevier, New York (2002)
12. Saely, C.H., et al.: Brown versus white adipose tissue: a mini-review. Gerontology (2012)
13. Salans, L.B., Cushman, S.W., Weismann, R.E.: Studies of human adipose tissue, adipose cell size and number in non obese and obese patients. J. Clin. Inv. **52**
14. Nascimento, C.M., Ribeiro, E.B., Oyama, L.M.: Metabolism and secretory function of white adipose tissue: effect of dietary fat. Anais da Academia Brasileira de Ciências **81**(3), 453–466 (2009)
15. Health risks of being Overweight, U.S. Department of Health and Human Services (2007)
16. Toss, F.: Body Fat Distribution, Inflammation and Cardiovascular Disease. Sweden (2011). ISSN: 0346-6612
17. Chen, C.H., Chen, Y.Y., Chuang, C.L., Chiang, L.M., Chiao, S.M. and Hsieh, K.C.: The study of anthropometric estimates in the visceral fat of healthy individuals. Nutr. J. (2014)

18. Peele, L.: Body fat percentage: a complete guide to evaluation and measurement. (2010)
19. Dey, D.K., Bosaeus, I.: Body composition estimated by bioelectric impedance in the swedish elderly—development of population—based prediction equation and reference values of fat—free mass and body fat for 70–75-year olds. Europ. J. Clin. Nutr
20. Fernandes, R.A., Rosa, C.S., Buonani, C., de Oliveira, A.R., Júnior, I.F.F.: The use of bioelectrical impedance to detect excess visceral and subcutaneous fat. Journal de Pediatria (2007)
21. Sun, S.S., et al.: Development of bioelectric impedance analysis prediction equations for body composition with the use of multicomponent model for use in epidemlogic surveys. Am. Soc. Clin. Nutr. **77**(2), 331–40
22. Zhong, L.L., Kun, W., et al.: The combination effects of body acupuncture and auricular acupressure compared to sham acupuncture for body weight control: study protocol for a randomized controlled trial. BioMed Cent. **17**(1), 346
23. Dehghan, M., et al.: Is bioelectric impedance accurate for use in large epidemilogical studies?. BioMed Cent. Nutr. J. (2008)
24. Hannah, J.G., Gladis, D.: An entropy based classification of body fat using fuzzy rules commingled with genetic algorithm. Int. J. Recent Technol. Eng. (IJRTE). **8**(3), 2493–2496 (2019). ISSN: 2277-3878
25. Deurenberg, P., et al.: Singaporean Chinese adolescents have more subcutaneous adipose tissue than Dutch Caucasians of the same age and body mass index. Asia Pacif. J. Clin. Nutr. **12**(3), 261–265 (2003)
26. www.mathworks.com/help/data_analysis/linear-regression.html
27. Kalyani, R.R., et al.: Age-related and disease related muscle mass: the effect of diabetes, obesity and other diseases. Lancet Diab. Endocr. **2**(10), 819–829 (2014)
28. Thein, H.T.T.: Island model based differential evolution algorithm for neural network training. Intl. J. Adv. Comput. Sci. 3(1), 7 (2014). ISSN: 2322-5157
29. Thein, H.T.T., Tun, K.M.M.: An analysis of migration in the island model on feed-forward neural network. Glob. J. Adv. Res. **1**(2), 222–231. ISSN: 2394-578

Design of Compact Super-Wideband Monopole Antenna for Spectrum Sensing Applications

Pawan Kumar, Shabana Urooj, and Bharat J. R. Sahu

Abstract A compact quasi-self-complementary super-wideband (SWB) monopole antenna excited by a tapered coplanar waveguide (CPW) feed line is presented in this paper. The size of the antenna is 31 mm × 26 mm × 1.6 mm, and it can operate in the range of 1.2–35 GHz ($S_{11} \leq -10$ dB). The WLAN and X-band based satellite communication signals are eliminated by introducing a complementary split-ring resonator (CSRR) in the radiator of the monopole antenna. The design and performance of the proposed antenna are validated by using the ANSYS HFSS® simulation tool. The peak gain is found to be 6 dB. The proposed antenna can be used in cognitive radios for spectrum sensing applications.

Keywords Cognitive radio · Self-complementary · SWB

1 Introduction

With the advancements in wireless communication systems, the use of ultra-wideband (UWB)/super-wideband (SWB) technology is increasing owing to its useful features such as small power consumption and high data rate transmission. After the Federal Communications Commission (FCC) released the regulations on UWB [1], a sharp surge has been noticed in UWB research for designing compact cheap antennas for various microwave applications [2, 3]. The UWB antenna

P. Kumar (✉) · S. Urooj
Department of Electrical Engineering, School of Engineering, Gautam Buddha University, Greater Noida 201308, India
e-mail: pawangupta.iitb@gmail.com

S. Urooj
Department of Electrical Engineering, College of Engineering, Princess Nourah Bint Abdulrahman University, Riyadh 84428, Kingdom of Saudi Arabia

B. J. R. Sahu
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be) University, Bhubaneswar 751030, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_23

possesses a bandwidth ratio of 10:1, and its bandwidth is defined from 3.1 to 10.6 GHz, while the SWB antenna exhibits more than 10:1 bandwidth ratio [4, 5].

In the literature, self-complementary antennas (SCA) have been reported as SWB/UWB antennas [6, 7]. The self-complementary SWB antenna can provide infinite bandwidth (under ideal conditions) if it is designed on an infinite length ground surface. The input impedance of the SCA structure has a constant value of 188.5 Ω, which is independent of the frequency used. But in real conditions, the antenna size is finite, and a balun/transition is required for converting impedance of 188.5–50 Ω, so that the antenna can be combined with standard RF components. To resolve these issues, a quasi-self-complementary antenna (QSCA) is proposed, which is a compact-sized configuration with a built-in impedance matching circuit [8–10].

Recently, the cloud radio access network (RAN) is suggested as the most suitable radio access technique. Due to its energy-efficient operation and low installation cost, it is considered as a promising candidate for the operation of next-generation radio/mobile base terminals. A high-performance cloud RAN needs an outstanding wideband RF front end, which is not possible without using extraordinary antenna structures. Therefore, state-of-the-art antenna structures with massive bandwidth and least dispersion are needed to fulfill the cloud RAN RF front end system specifications [11].

In this paper, a compact SWB QSCA structure possessing a very large bandwidth of 1.2–35 GHz and dual band-notched characteristics is proposed. A tapered coplanar waveguide (CPW) feed line is integrated with the elliptical monopole to match 50 Ω impedance of the SMA connector. The WLAN (5.5 GHz) and satellite communication (7.25–7.75 GHz/7.9–8.4 GHz) bands are eliminated by introducing a complementary split-ring resonator (CSRR) in the resonating patch of the monopole antenna. The work in the paper is organized as follows: Section 2 gives details of the SWB antenna design. Section 3 gives insight into the results and their discussion. Section 4 concludes the characteristics and advantages of the designed antenna.

2 Antenna Configuration

2.1 SWB QSCA

The layout of the designed SWB antenna is presented in Fig. 1. The antenna is constructed on FR-4 dielectric substrate (with relative permittivity 4.4 and loss tangent 0.001) of size 31 mm × 26 mm × 1.6 mm. The antenna consists of a semi-elliptical monopole patch integrated with a tapered CPW feed line. The tapered line provides an impedance matching between the elliptical radiator and the standard 50 Ω SMA connector end.

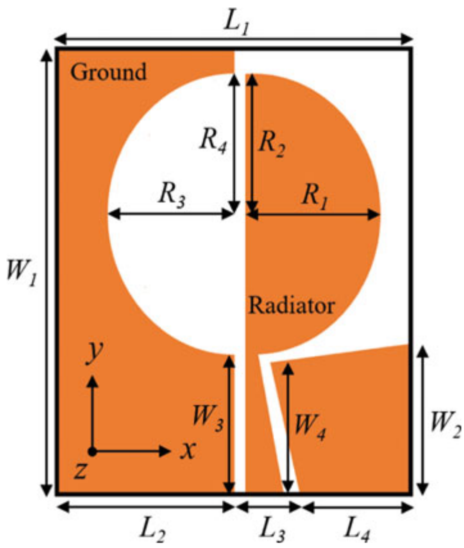
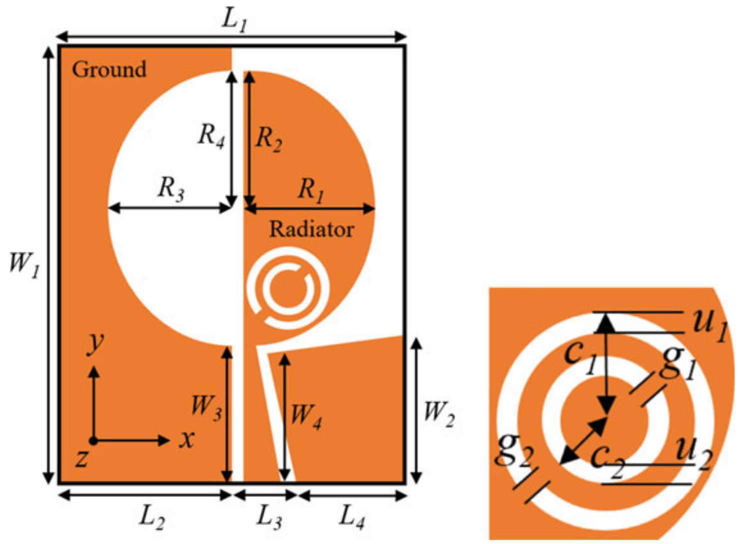


Fig. 1 Layout of the printed SWB antenna

A semi-elliptical slot analogous to the resonating patch is introduced in the coplanar ground plane of the monopole antenna to obtain self-complementary geometry. The dimensions (R_1, R_2, R_3, and R_4) of the elliptical radiator and the complementing slot are optimized for high eccentricity to excite higher modes. Furthermore, the dimensions of the radiator and the complementing slot are not identical, and therefore, the layout of the designed SCA structure is not perfectly elliptical and characterized as quasi-elliptical. The optimized dimensions of the designed SWB antenna are listed in Table 1.

Table 1 Dimensions of the designed SWB antenna

Parameters	Values (mm)	Parameters	Values (mm)
L_1	26	R_2	10.2
L_2	12	R_3	7.5
L_3	5	R_4	10.2
L_4	9	u_1	0.5
W_1	31	u_2	0.5
W_2	10	c_1	2.8
W_3	9.6	c_2	1.6
W_4	8.7	g_1	0.5



(a) (b)

Fig. 2 Dual band-rejected SWB antenna **a** layout **b** CSRR

2.2 QSCA with Notch Bands

Figure 2a displays the layout of the designed dual band-notched printed SWB antenna. The band elimination characteristics at WLAN (5.5 GHz) and satellite communication (7.25–7.75 GHz/7.9–8.4 GHz) bands are realized by introducing a CSRR in the resonating patch of the elliptical monopole antenna. The center frequency of the notch band is considered as [12]

$$f_{r_i} = \frac{v}{2\pi c_i \sqrt{\varepsilon_{\mathrm{r_{eff}}}}}; \quad i = 1, 2 \tag{1}$$

$$\varepsilon_{r_{\mathrm{eff}}} = \frac{\varepsilon_r + 1}{2} \tag{2}$$

where ε_{r} is the relative permittivity of the dielectric substrate, v is the velocity of light in free space, and c_i is the radius of outer and inner rings. The details of the CSRR dimensions are given in Fig. 2b.

3 Results Discussion

Figure 3 displays three configurations of the SWB antenna, where the first geometry illustrates an elliptical-shaped monopole antenna covering ($S_{11} \leq -10$ dB) bandwidth of 1.2–35 GHz.

In the second geometry, a split-ring is introduced in the radiating patch of the SWB antenna. The split-ring disturbs the current pattern of the antenna, thereby eliminating the 5.5 GHz (WLAN) band from SWB. Further, one more split-ring (complementary to the first split-ring) is introduced in the patch of SWB antenna, which notches the satellite transmitting and receiving (7.25–7.75 GHz/7.9–8.4 GHz) bands. The simulated S_{11} of the SWB antenna with/without notch bands are presented in Fig. 3.

The surface current distributions at 5.5 GHz and 7.8 GHz are displayed in Figs. 4a, b, respectively. At 5.5 GHz, the maximum current exists nearby the outer ring of the CSRR, whereas at 7.8 GHz, it is maximum around the inner ring of the CSRR. The simulated gain curve of the band-rejected SWB antenna is displayed in Fig. 5. A rapid fall in the antenna gain is noticed at two notch band frequencies. The co-polarized and cross-polarized radiation patterns of the SWB antenna are displayed in Fig. 6. SWB monopole antenna illustrates almost omnidirectional behavior in the H-plane and nearly bidirectional behavior in the E-plane.

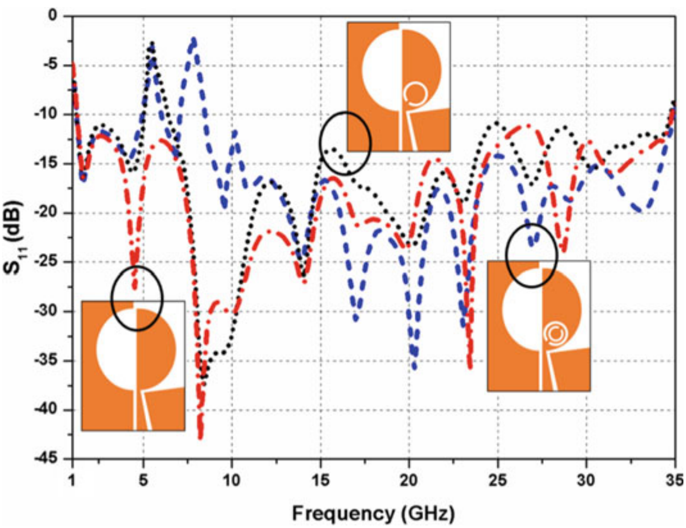
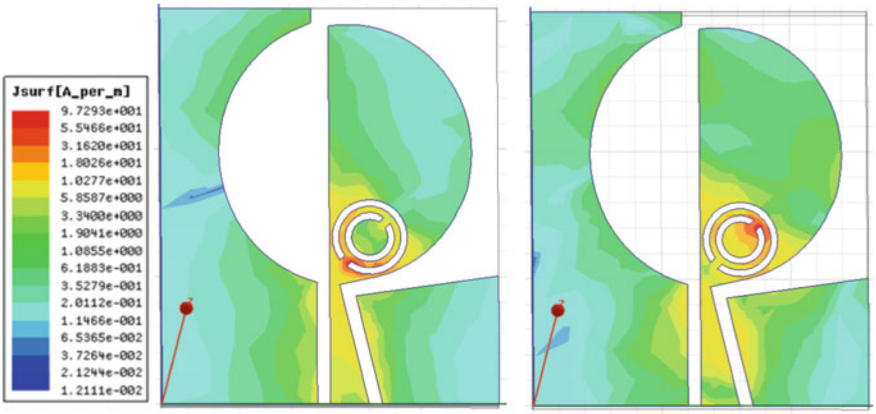


Fig. 3 Simulated S_{11} of the proposed SWB antenna with/without notch bands



(a) (b)

Fig. 4 Surface current distribution at **a** 5.5 GHz and **b** 7.8 GHz

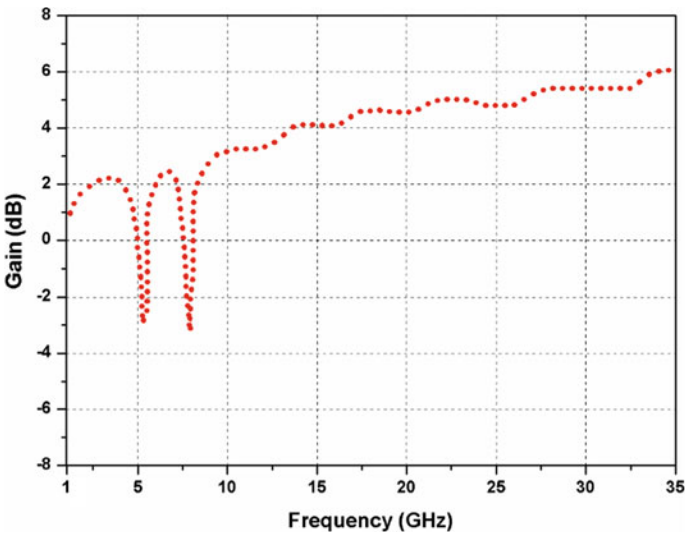


Fig. 5 Simulated gain of the proposed dual band-rejected SWB antenna

4 Conclusion

A small size self-complementary SWB antenna with dual band rejection (WLAN and satellite communication bands) features is investigated. The designed SCA consists of an elliptical resonating patch and a matching slot (which is a complement of elliptical shape), both present on the same side of the material substrate. The SWB is achieved due to the quasi-elliptical geometry of the designed monopole antenna.

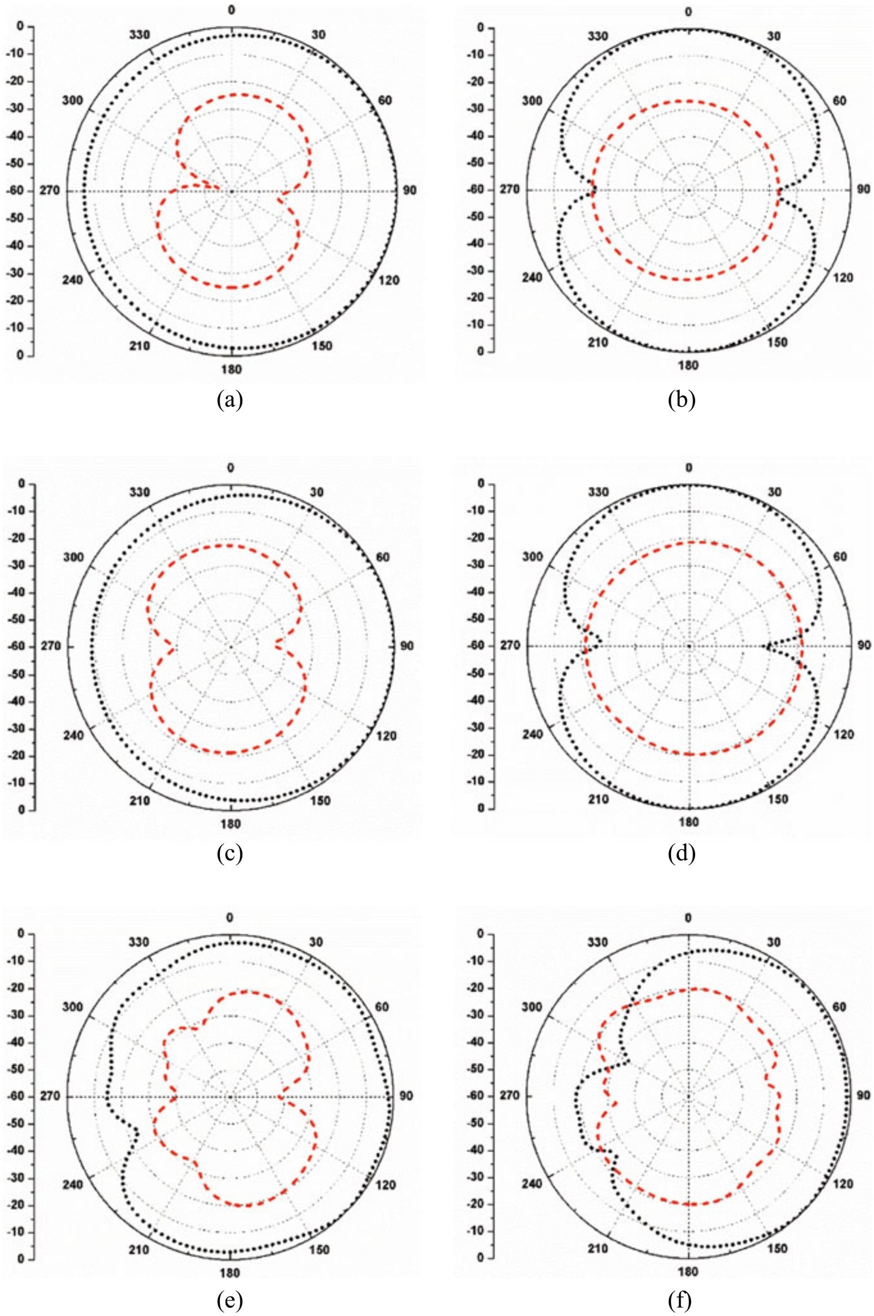


Fig. 6 Simulated patterns of the printed SWB antenna **a** 4 GHz, H-plane; **b** 4 GHz, E-plane; **c** 9 GHz, H-plane; **d** 9 GHz, E-plane; **e** 14 GHz, H-plane; **f** 14 GHz, E-plane (co-polar: black; cross-polar: red)

The printed antenna offers a bandwidth ratio of 29:1, the peak gain of 6 dB, and omnidirectional radiation performance in the working frequency range. The radiation results demonstrate that the designed antenna shall be useful for cognitive radio spectrum sensing applications.

References

1. Federal Communications Commission: First report and order, revision of part 15 of the commission's rules regarding ultra wideband transmission systems. Washington, DC (2002)
2. Biswas, B., Ghatak, R., Poddar, D.R.: A fern fractal leaf inspired wideband antipodal Vivaldi antenna for microwave imaging system. IEEE Trans. Antennas Propag. **65**(11), 6126–6129 (2017)
3. Pazin, L., Leviatan, Y.: Reconfigurable rotated-T slot antenna for cognitive radio systems. IEEE Trans. Antennas Propag. **62**(5), 2382–2387 (2014)
4. Yan, S., Soh, P.J., Vandenbosch, G.A.E.: Wearable ultra wideband technology—a review of ultrawideband antennas, propagation channels, and applications in wireless body area networks. IEEE Access **6**, 42177–42185 (2018)
5. Okas, P., Sharma, A., Das, G., Gangwar, R.K.: Elliptical slot loaded partially segmented circular monopole antenna for super wideband application. AEU Int. J. Electron. Commun. **88**, 63–69 (2018)
6. Mushiake, Y.: Self-complementary antennas. IEEE Antennas Propag. Mag. **34**(6), 23–29 (1992)
7. Mushiake, Y.: A report on Japanese development of antennas: from the Yagi-Uda antenna to self-complementary antennas. IEEE Antennas Propag. Mag. **46**(4), 47–60 (2004)
8. Trinh-Van, S., Kwon, G., Hwang, K.C.: Planar super-wideband loop antenna with asymmetric coplanar strip feed. Electron. Lett. **52**(2), 96–98 (2016)
9. Lin, C.-C., Huang, C.-Y., Su, J.-Y.: Ultra-wideband quasi-self-complementary antenna with band-rejection capability. IET Microw. Antennas Propag. **5**(13), 1613–1618 (2011)
10. Guo, L., Wang, S., Gao, Y., Wang, Z., Chen, X., Parini, C.G.: Study of printed quasi-self-complementary antenna for ultra-wideband systems. Electron. Lett. **44**(8), 511–512 (2008)
11. Checko, A., Christiansen, H.L., Yan, Y., Scolari, L., Kardaras, G., Berger, M.S., Dittmann, L.: Cloud RAN for mobile networks—a technology overview. IEEE Commun. Surv. Tuts. **17**(1), 405–426 (2015)
12. Almalkawi, M.J., Devabhaktuni, V.K.: Quad band-notched UWB antenna compatible with WiMAX/INSAT/lower/upper-WLAN applications. Electron. Lett. **47**(19), 1062–1063 (2011)

Probabilistic and Distance-Aware Clustering Approach to Decrease the Effect of Hot Spot in WSN

Ruchi Kulshrestha and Prakash Ramani

Abstract Wireless sensor network has various application areas like dense forest fire detection, underwater surveillance, habitat monitoring, healthcare, agriculture, etc. There are some challenges associated with wireless sensor networks: designing of an energy-efficient algorithm, deployment and localization of sensors, data security, synchronization, etc. Some applications of WSN are constrained to be energy efficient as battery replacement at sensor nodes is not possible. Clustering is an effective method for reduced energy consumption at sensor nodes. In clustering, clusters are made by grouping sensor nodes; then, a node is elected as cluster head that is responsible for data transmission to the destination. Cluster heads that are near to base station have more workload hence become dead early. Soon an area generated with all dead nodes around the base station and further transmission of data becomes blocked. It is called hot-spot effect. In this paper, the problem of hot-spot impact is addressed, and a hybrid of probabilistic and distance-aware approach-based energy-efficient algorithm is proposed to decrease the hot-spot effect. In proposed protocol, the distance of nodes from the sink is considered in selection of nodes as cluster head. Sensor nodes within the area of optimal distance from the sink are not elected as cluster head; thus, it results in less energy consumption at these nodes. Results are simulated in MATLAB. Analysis of results and comparison has been done with LEACH that is a well-known existing energy-efficient clustering protocol. Result analysis shows that proposed protocol gives better results in terms of network lifetime.

Keywords WSN · Clustering · LEACH · Hot spot · Energy efficiency · Network lifetime

R. Kulshrestha (✉) · P. Ramani
Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, India
e-mail: ruchi.kul23@gmail.com

P. Ramani
e-mail: prakash.ramani@jaipur.manipal.edu

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_24

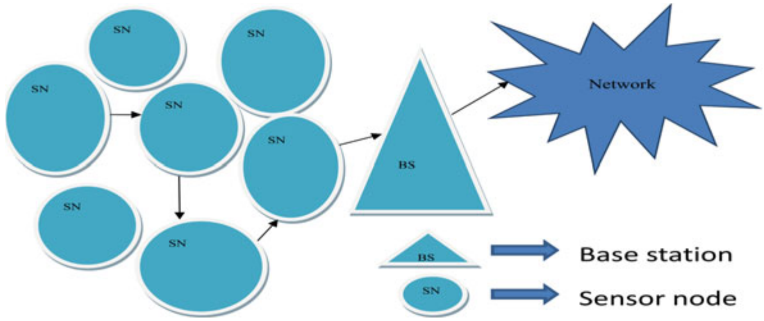


Fig. 1 WSN architecture consists of sensor nodes, base station, and network

1 Introduction

1.1 Wireless Sensor Network

Wireless sensor network (WSN) is a group of sensors that are interconnected for communication. Sensors are small devices that are used for sensing physical attributes of the external environment and sending that data to the required device. That sensed data is further used for analysis and to take some required actions. Wireless sensor network has vast application areas.

1.2 WSN Challenges

Wireless sensor networks have some challenges:

(1) Limited energy resources
(2) Security
(3) Time synchronization
(4) Fault tolerance
(5) Responsiveness
(6) Sensors deployment (Fig. 1).

1.3 Energy Efficiency

Some applications of WSNs like dense forest fire detection, military applications, habitat monitoring, etc., are constrained to be energy efficient as battery replacement is not possible. So for such applications, protocols and algorithms should be energy efficient. Energy efficiency can be achieved by making a system that results in low energy consumption.

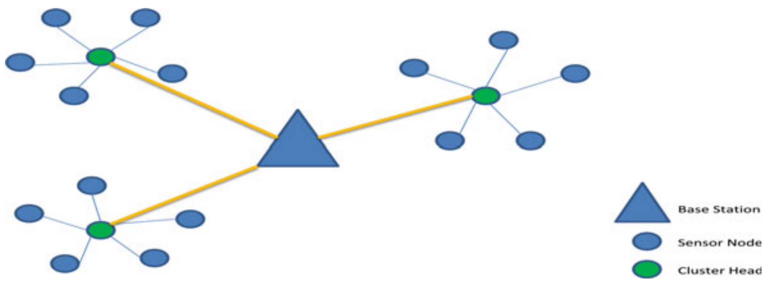


Fig. 2 Clusters are made up by grouping of sensor nodes. Cluster head is responsible for transmitting data to base station

1.4 Clustering

Clustering is a method in WSN to achieve energy efficiency and to prolong the network lifetime. In clustering, the whole sensing field is divided into groups of sensors called clusters. Each cluster has an elected cluster head, which is responsible for aggregation and transmission of data coming from respective cluster members (Fig. 2).

1.5 LEACH

Low-energy adaptive clustering hierarchy (LEACH) is a standard clustering protocol based on the probabilistic approach. It has two phases: setup and steady phase [1]. A probabilistic approach is used to select the cluster head among all sensor nodes [2]. Cluster head advertises itself as a cluster head through broadcasting its ID. Both TDMA and CDMA are used in clustering and data transmission (Fig. 3).

2 Distance Effect

During the deployment of sensors, all nodes are scattered throughout the sensing field following any suitable deployment strategy. Distance between sensor nodes and the base station is an essential parameter for energy consumption. Energy dissipation during data transmission is less for the nodes that are near to base station as data have to travel less distance.

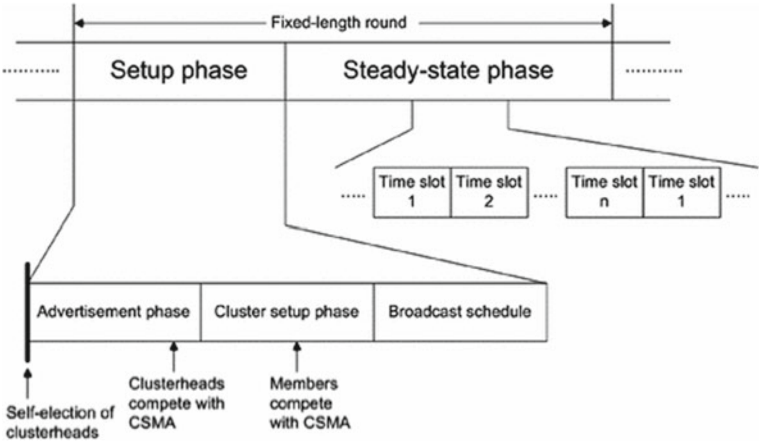


Fig. 3 LEACH phases [3]

2.1 *Hot-Spot Effect*

In clustered WSN, each cluster has its cluster head. All cluster heads have to transmit data coming from its cluster members to the base station. In multi-hop routing, cluster heads that are near to base station have more workload of data transmission. It results in frequent energy dissipation. And these nodes become dead early. Soon an area is created, having all dead nodes near the base station; this is called hot spot. Due to this effect, data transmission got stranded.

2.2 *Residual-Energy-Based Clustering*

Residual energy is available energy at any node. If a node is being elected as cluster head, then it has to consume more energy in data aggregation and transmission, so the nodes having more residual energy should be selected as cluster heads. Residual-energy-based selection of cluster head will help to diminish the hot-spot effect as hot-spot effect results in faster drainage of energy at the cluster heads that are nearer to sink.

3 Related Work

Nodes having more residual energy should be elected as CH to prolong the network lifetime. Li et al. [4] proposed a protocol called distributed energy-efficient clustering (DEEC). It is for heterogeneous wireless sensor networks and is based on residual energy of nodes. In DEEC, selection of cluster head confides in the remaining energy of nodes and the equate energy of the network. This is for the heterogeneous network where some nodes have more initial energies than others. All nodes are divided into types: advanced nodes and normal nodes. Nodes having more initial energy are called as advanced nodes, and these have more chances to become cluster head, and rest are called as normal nodes. Liu et al. [5] proposed a protocol called energy-aware routing protocol (EAP). Nodes exchange the information of residual energy with its neighbor nodes. A table is formed at each node regarding the residual energy of neighboring nodes. And this table gets updated. Every node of a cluster considers the broadcasting delay as contending factor to become CH. It is also based on residual energy and the average energy of the network. Chen et al. [6] proposed a protocol called QoS-based adaptive clustering (QAC). In QAC algorithm, there are master CH and slave CH. Some provisional CHs are allotted as master CHs. In this, a threshold value of a number of nodes is considered and used for the slave CH election by master–slave. Other member nodes of the cluster can select cluster head from master CH and slave CH. Master CH and slave CH can relocate their functions if any out of them become dead. Ali et al. [7] proposed a protocol called LESCA. It is based on spectral classification. It considers residual energy and distance of a node to the base station to select CH. As the distance between nodes and the base station is a very significant factor for energy consumption, these distances are considered as parameters. Santosh et al. [8] proposed a protocol called EECPEP-HWSN. In this, selection of CH is based on current adaptable energy, introductory energy, and hop count from the base station. Also, internal overheads are getting reduced to a greater extent with the initial division of network in sections. Due to this, formed clusters have controlled size.

Re-clustering consumes some amount of energy. It can be saved by static clustering. Zahemati et al. [9] proposed a protocol called energy-efficient protocol with static clustering (EEPSC). It is based on a static clustering scheme. The whole process has various rounds. At every round, there are three phases. The setup phase is for network subdivision. The base station makes the clusters. BS broadcasts a message to all nodes with an ID of wished for CHs in the network. Nodes join their respective CHs. This scheme is energy efficient as it does not perform re-clustering. CH election is done on energy strength of nodes. This also has some flaws like scaling is not possible, as clusters are static. Cluster head near the base station is busier than other cluster heads. If cluster size near the base station is small, then load on cluster head can be balanced. Jianget al. [10] proposed a protocol called energy-balanced unequal clustering (EBUC). This protocol arranges unequal clusters. Clusters are unbalanced in a number of nodes in it. For cluster head selection, energy of nodes is considered, and particle swarm optimization is applied for this at base station. Size of clusters is small near base station to decrease the effect of hot spot. Node's distance from cluster head and cluster size can be considered in clustering to achieve energy efficiency and prolonging network lifetime. Vipin et al. [11] proposed balanced clustering. It has two phases: setup and steady phase. It works on two parameters ThCluster and ThDistance. Any node can join the CH if it has members < Thcluster and distance < ThDistance. During rescue phase, if above conditions do not match, then node will join the nearest node. This approach extends the network lifetime and lowers the node death rate. As LEACH uses the probabilistic function to select CH, some energy consumed due to CH rotation. Muhammad et al. [12] proposed a protocol named EE-MRP. It uses multistage data transmission. For creating multistage transmission, the entire network is sectioned into various stages containing optimal number of cluster heads. As every stage has approx equal number of cluster heads, network lifetime and performance are increased. Static cluster head selection is used to minimize energy consumption.

4 Proposed Protocol

In this paper, we proposed a distance-based probabilistic clustering protocol. A threshold value of an optimal distance between node and sink is considered. If a node is situated under the optimal distance, then it will be as forwarder node and will not participate in clustering. Cluster heads of all clusters send their data to these nodes to be transmitted to the base station. And rest of nodes participates in clustering based on probabilistic approach of LEACH protocol (Fig. 4).

Algorithm

Step 1: Random deployment of sensors in the sensing field.
Step 2: Consider an optimal distance Opt_Dist. For i=1 to n(number of nodes), Repeat step 3 & 4.
Step 3: Calculate distance d of a node from the sink.
Step 4: If (d<=Opt_Dist)
Flag(i)=1,
else Flag(i)=0
Step 5: Probabilistic Approach based clustering(LEACH based) for all nodes if Flag(i)=0
Step 6: End

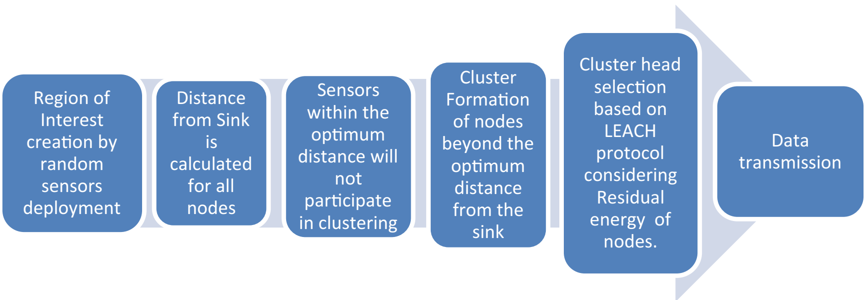


Fig. 4 Sequential flow of proposed protocol

5 Results

Here, E_0 is initial energy, E_{fs} is energy dissipation in free space, E_{mp} is energy dissipation in multipath transmission, and EDA is energy consumption for data aggregation.

Simulations have been performed in MATLAB to obtain results. Table 1 shows the simulation parameters. MATLAB simulations are done for total nodes 100 and 200 with 5000 iterations or number of rounds. Tables 2 and 3 show network lifetime comparison between LEACH and proposed protocol. Comparison parameters are First Node Dead (FND) and Last Node Dead (LND) at the round no. of algorithm. Latest FND and LND present an increased lifetime of the network. Figures 5 and 6

Table 1 Simulation parameters and respective values in MATLAB

Simulation parameters	Value
Total rounds	5000
Total nodes	100, 200
EDA	0.5
E_0	0.5 J
E_{fs}	10 pJ
E_{mp}	0.0013 pJ

Table 2 Comparative results for LEACH and proposed (for 100 nodes)

	LEACH	Proposed
First node dead at round (FND)	997	1236
Last node dead at round (LND)	3265	4998

Table 3 Comparative results for LEACH and proposed (for 200 nodes)

	LEACH	Proposed
First node dead at round (FND)	992	1299
Last node dead at round (LND)	3708	4999

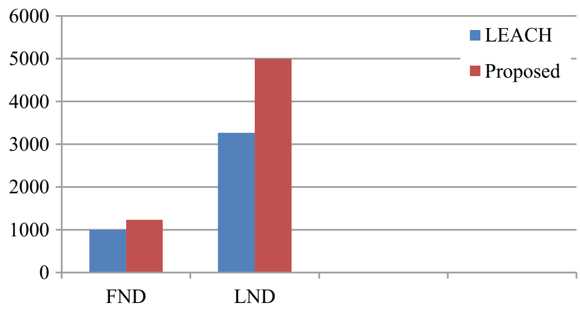


Fig. 5 Comparative analysis of network lifetime between LEACH and proposed (For $n = 100$)

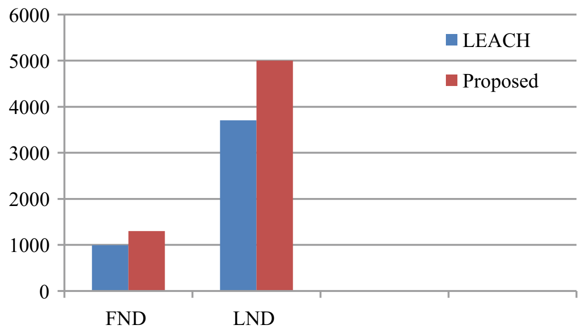


Fig. 6 Comparative analysis of network lifetime between LEACH and proposed (For $n = 200$)

show the lifetime comparison of LEACH and proposed protocol. The figure clearly shows that the proposed protocol has the latest FND and LND, so a lifetime of network is increased, which results in the energy-efficient network.

6 Conclusion

Wireless sensor network has vast application areas. Some applications require energy efficiency in working. Clustering is one of the approaches to achieve energy efficiency in the system. LEACH is one of the classical methods of clustering. Sufficient work has been done in this area; still, some challenges are in existence. The hot-spot effect is one of the problems that require consideration. In this paper, a hybrid approach of probabilistic and distance-aware approach is discussed. Simulation has been done in MATLAB. Comparative results of network lifetime between LEACH and proposed method are shown. The analytic study of results shows that the proposed method gives better output in terms of network lifetime. Proposed approach tries to decrease the effect of hot spot in the WSN.

References

1. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd annual Hawaii international conference on system sciences. IEEE, 10pp, (2000)
2. Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. Wirel. Commun. **1**(4), 660–670 (2002)
3. Karl, H., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. Wiley, New Jersey (2007)
4. Qing, L., Zhu, Q., Wang, M.: Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. Comput. Commun. **29**(12), 2230–2237 (2006)
5. Liu, M., Cao, J., Chen, G., & Wang, X. (2009). An energy-aware routing protocol in wireless sensor networks. Sensors **9**(1), 445–462. (2009)

6. Chen, W., Li, W., Shou, H., Yuan, B.: A QoS-based adaptive clustering algorithm for wireless sensor networks. In: 2006 International Conference on Mechatronics and Automation. IEEE, pp. 1947–1952 (2006)
7. Jorio, A., El Fkihi, S., Elbhiri, B., Aboutajdine, D.: An energy-efficient clustering routing algorithm based on geographic position and residual energy for wireless sensor network. J. Comput. Netw. Commun. (2015)
8. Purkar, S.V., Deshpande, R.S.: Energy efficient clustering protocol to enhance performance of heterogeneous wireless sensor network: EECPEP-HWSN. J. Comput. Netw. Commun. (2018)
9. Zahmati, A.S., Abolhassani, B., Shirazi, A.A.B., Bakhtiari, A.S.: An energy-efficient protocol with static clustering for wireless sensor networks. Int. J. Electron. Circ. Syst. **1**(2), 135–138 (2007)
10. Jiang, C.J., Shi, W.R., Tang, X.L.: Energy-balanced unequal clustering protocol for wireless sensor networks. J. China Univ. Posts Telecommun. **17**(4), 94–99 (2010)
11. Pal, V., Singh, G., Yadav, R.P.: Balanced cluster size solution to extend lifetime of wireless sensor networks. IEEE Inter. Things J. **2**(5), 399–401 (2015)
12. Khan, M.K., Shiraz, M., Zrar Ghafoor, K., Khan, S., Safaa Sadiq, A., Ahmed, G.: EE-MRP: energy-efficient multistage routing protocol for wireless sensor networks. Wirel. Commun. Mob. Comput. (2018)

A Real-Time Sentiments Analysis System Using Twitter Data

Aditya Dave, Santosh Bharti, Samir Patel, and Sambit Kumar Mishra

Abstract As social media platforms become the go-to for knee-jerk reactions on events by the current populous, it has become extremely important for event managers, celebrities, and organizations to constantly monitor their perceived social image online. This becomes especially difficult during key periods of heightened activity, like events, announcements, etc. As the rate at which the tweets are posted is much higher than what a human can read or comprehend. In this paper, we exploit existing sentiment analysis techniques to develop a real-time sentiment analysis system that provides us with real-time sentiments of the audience on the micro-blogging site, Twitter, toward an event, organization, or person. This system serves to act as a feedback mechanism helping the users to understand, the perceived image of the event/organization. This feedback, if provided in a timely manner, can be used to improve the situation at hand or act as a positive reinforcement for the team. In today's world, neglecting social media can prove detrimental to the success of an event or organization. We analyze two different events from two separate domains to understand and demonstrate the benefits of our system.

Keywords Real-time system · Real-time tweets · Sentiment analysis

A. Dave (✉) · S. Bharti · S. Patel
Pandit Deendayal Petroleum University, Gandhinagar, Gujarat, India
e-mail: aditya.dict16@sot.pdpu.ac.in

S. Bharti
e-mail: sbharti1984@gmail.com

S. Patel
e-mail: samir.patel@sot.pdpu.ac.in

S. K. Mishra
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan (Deemed to Be) University, Bhubaneswar, India
e-mail: sambitmishra@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_25

1 Introduction

In this modern age, it has become imperative for event managers/event organizers to always be aware of the perceived image of their event. The easiest way in which they can do this is via constantly monitoring their social media pages and recognizing what the masses are feeling, and this is a very cumbersome task. As messages and tweets can be posted at a rate much faster than any human can read or comprehend them, making it impossible to monitor the sentiments of the masses in real-time. In this paper, we have developed a system that can analyze such messages, from the micro-blogging site, Twitter, and provide live feedback to the user about the current sentiments of the masses [1].

This system can be used as a measuring stick for people's emotion, as the event progresses. It can also be used to see in real-time the emotions each new incident evokes. This feedback can be used to adapt the proceedings of the event for a favorable response, in real-time. There have been a lot of advancements in the recent years in the field of sentimental analysis, but most, if not all, of these systems require a storage medium to first capture the data and then process it. A real-time streaming system eliminates this. Our system can function independently and does not require a long-term storage medium, although a storage medium can be added if needed for deeper analysis. As social media web sites continue to grow, managing the amount of data generated by them becomes more and more cumbersome leading to higher costs and management challenges. Leveraging the capabilities of modern high performance multi-threaded processors, our system uses multiple threads to fetch data and process it in real-time to display an accurate result.

Our system provides an instantaneous albeit a coarser response that can help a user to correlate the current situation with the online sentiment. This is different from conventional systems where the delay due to processing is very high making it impossible to accurately pin-point the incident affecting the sentiment. In this paper, we have outlined the use of this system as a real-time feedback system for events, but this work can also be used to monitor a person's or organization's social image by fine tuning the track parameters of the Tweepy listener, to capture tweets that only pertain to one person or organization.

The paper is structured as follows: Sect. 2 serves to give an idea of similar kind of systems that have already been implemented in this field. In the next section (Sect. 3), we provide a detailed model of our systems and layout the different features of the same. In Sect. 4, we look upon two case studies to evaluate our systems. In Sect. 5, we write our closing thoughts and outline the future scope for our system.

2 Literature Survey

In the last decade, a lot of advancements have been made in the field of sentimental analysis, due to the immense increase in the computational power available and because of increase in the amount of social media data that is available. This fact is depicted by Pak et al. [2], where they build a sentiment analyzer using a micro-blogging site as their corpus.

But even after extensive research, we could not find such varied results for a real-time system that analyzed tweets and calculated the sentiments of the tweets. As per my best knowledge there is only one paper that tried political sentimental analysis in real-time to establish a relation between expressed public sentiment and electoral events [3]. This paper though does not provide a means to accurately connect the sentiments of the masses toward one event in real-time. This paper focuses more on the broader picture, encompassing the entire political campaigning phase.

3 Proposed System

In this section, we have a detailed look over our proposed system to analyze Twitter data in real-time. An overview of the same is shown in Fig. 1.

3.1 Tweepy Stream Listener

Tweepy is a very widely used Python library to interface with the Twitter API [4]. This library handles the OAuth authentication expected by Twitter and provides an object that can be used to access the Twitter API. In this paper, Tweepy was used to create a streamer object on particular keywords. This object streams all the tweets that contain or "mention" the keywords. This is done using the track property of the streaming API [5].

After fetching a tweet, the object calls a callback function that can be used for further processing. Twitter has severe rate limits on the free API defined by a status code 420 [6]. In this paper, every time a rate limit is encountered a cool down period of 60 s is set after which the program reconnects and resumes previous operation.

3.2 Preprocessing

Fairly simple preprocessing is carried out on all the tweets in order to ensure that the sentiment engine can effectively classify all the tweets correctly. All the tweets were stripped off the return handles (RT @xxx), URLs, Twitter handles (@xxx), and all other numeric and special characters except the '\#' symbol. An example of this is shown in Fig. 2.

The Twitter API does not always provide us with the completed tweet, but sometimes turns on the truncated flag and sends the complete tweet text as a separate property deep in the json structure [7] to ensure that we always have complete text

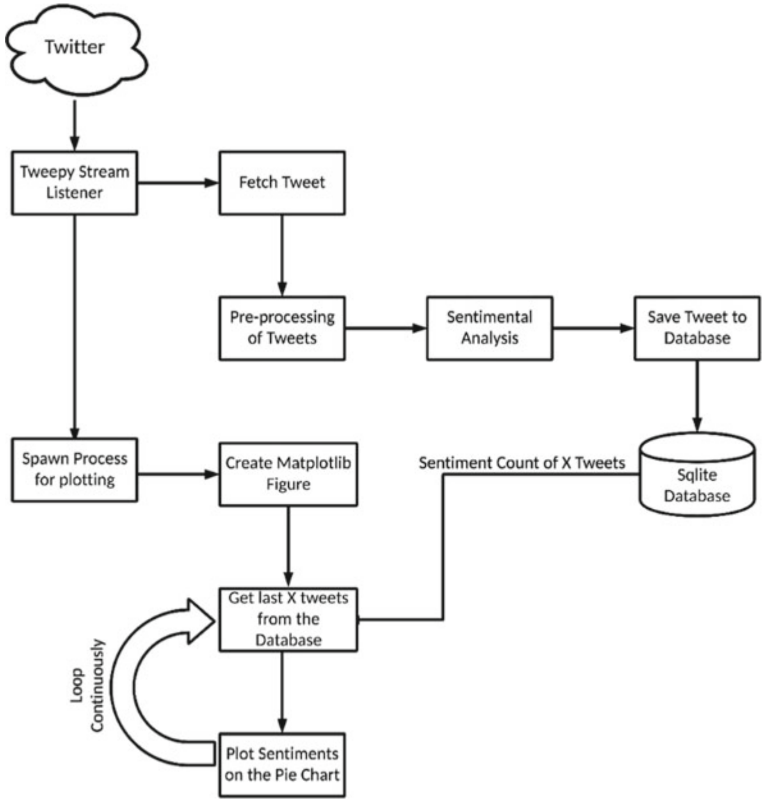


Fig. 1 A block diagram of the proposed system

Original Tweet: RT @FAP_Pelzeri: Verstappen's way to good results this season #F1 https://t.co/FHQeNFnOp7

Preporocessed: Verstappen's way to good results this season #F1

Fig. 2 An example of a raw tweet and a processed tweet

available to us that we check for the truncated flag and then retrieve the complete tweets from the "full_text" property in the extended text sub-tree.

3.3 Sentimental Analysis

We used a sentimental analysis technique known as "Valence Aware Dictionary and sEntiment Reasoner" or colloquially known as "VADER Sentiment Analysis" [8] that is part of the NLTK python library and is tuned to work on social media datasets.

It is a lexicon and rule-based analysis tool that does not require any training data and is fast enough to be used on streaming data. This library assigns positive, negative, and neutral sentiment to each lexicon of the statement and calculates a compound score using the sum of each individual lexicon rating. This compound score represents the overall sentiment of the statement. The standard threshold values on the compound score are used to classify the statement as positive, negative, or neutral to yield a compound score between -1 and $+1$.

3.4 Representation

In order to offer a visualization of the current twitter sentiment on the topic, the main process launches a sub-process that fetches last X number tweets and plots them on a pie chart using the matplotlib library [9]. The number of tweets that are considered can be changed, and this enables us to look at a bigger picture of the event or look at individual incidents that occur during the event. This feature also ensures that the graph never plateaus giving an accurate sentiment for all individual incidents.

3.5 Tweet Storage

After the tweet has been classified as positive/negative/neutral, it is saved in a sqlite3 database with its corresponding statement time and ID. This is a completely optional step, as the system can function by using inter process communication techniques to transfer data between the streaming and visualization process.

4 Case Studies

In this section, we look at two case studies that can corroborate our claims. The first case study analyzes the qualifying event of the Spanish Grand Prix that took place on Barcelona's circuit de Catalunya on May 11, 2019. The second case study looks at the sentiments toward the Indian National Congress (INC) party during the Indian Election Results on May 23, 2019. We focus specifically on the situation just before it was apparent that Rahul Gandhi will lose in Amethi.

4.1 Qualifying 3—Spanish Grand Prix 2019

In order to analyze the instantaneous reactions of the viewers on Twitter, we consider the last 75 tweets. This is to ensure that the sentiments do not saturate over time, and can also provide the instantaneous reaction of the people to the event and not the cumulative feeling towards the entire event. We picked the last 75 tweets as we noticed that after approximately 50 tweets the incident that was being tweeted about changed. We attributed this to the fact that motorsports is a very fast-paced sport and situations change quite dramatically within a span of a few seconds, and therefore, it is reasonable to assume that the current incident being tweeted about also changes quickly.

Figure 3a refers to the sentiments that were shown as soon as Valtteri Bottas set the fastest lap at 1:15.406 taking the pole position [10]. Fig. 3b shows the situation 11 s after the Bottas crosses the line, and here, we can see there is a rise of positive sentiment as folks on Twitter start appreciating and congratulating Bottas on his lap. Figure 3c shows the situation 37 s after Bottas crosses the line, and here, we can see an increase in the neutral sentiment. This is because of news organizations as well as teams start posting results and timing data. As these statements have no emotions and refer only to the statistics, they serve to increase the neutral share.

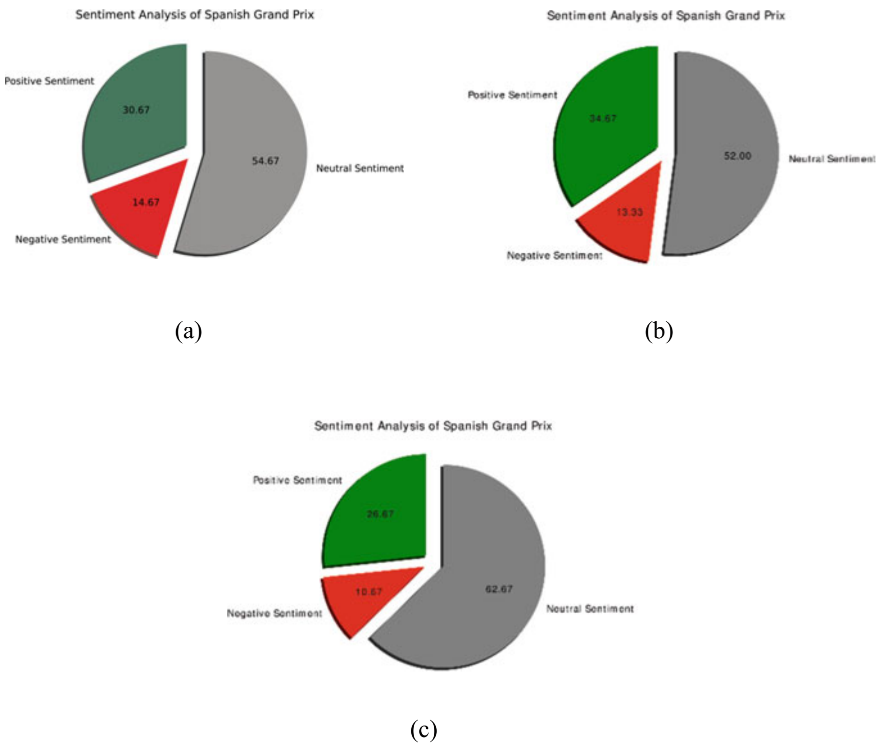


Fig. 3 Sentiments during the qualifying for the Spanish Grand Prix

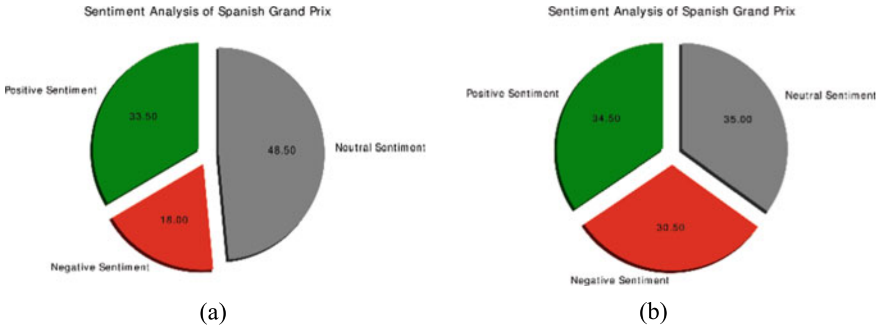


Fig. 4 Sentiments before and after Amethi seat announcement

4.2 Indian Election Results—2019

We used our system to show us the instantaneous sentiments toward the INC party during the morning of May 23, 2019.

The initial situation is depicted in Fig. 4a. As it becomes apparent that INC party will be facing a loss and Rahul Gandhi will lose to Smriti Irani in his home seat of Amethi, we find a drastic rise in the negative sentiment. This is accompanied by the fact that certain other key ministers, of the party, were trailing behind at that time. In the next Fig. 4b, as expected, we see a greater rise in the negative sentiment.

5 Conclusion and Future Scope

Using these case studies, we can say that our system can effectively predict the current sentiment of the masses towards different situations that arise during an event. In future, we can improve our system by providing a Web GUI for better representations and the ability to change the number of tweets being considered on the fly. We can further improve the speed as well as volume of the streaming API by using the enterprise Twitter API that does not feature a rate limit. As the system is highly modular, we can use a better sentiment detection algorithm that can detect more complex emotions or use emoticons and other special characters for classification, which can further increase our system's effectiveness.

References

1. Sela, R.: What you need to know about social media sentiment analysis. https://curatti.com/social-media-sentiment-analysis/. Last Accessed 29 May 2018
2. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. Proc. LREC **10** (2010)

3. Wang, H., et al.: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 115–120 (2012)
4. Tweepy. https://www.tweepy.org/. Last Accessed 14 May 2018
5. Standard stream parameters—Twitter Developers. https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/basic-stream-parameters.html. Last Accessed 14 May 2018
6. Response Codes—Twitter Developers. https://developer.twitter.com/en/docs/basics/response-codes.html. Last Accessed 14 May 2018
7. Tweet object—Twitter Developers. https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.html. Last Accessed 15 May 2018
8. Hutto, C.J., Gilbert, E.E.: VADER: a parsimonious rule-based model for sentiment analysis of social media text. In: 8th International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014 (2014)
9. Hunter, J.D.: Matplotlib: a 2D graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007)
10. Standings. https://www.formula1.com/en/results.html/2019/races/1004/spain/qualifying.html. Last Accessed 13 May 2018

Low-Profile Circularly Polarized Antenna for Contemporary Wireless Communication Applications

Bharat J. R. Sahu, Sachin Kumar, and Shobhit Saxena

Abstract A compact-sized, low-contour, microstrip line-fed circularly polarized (CP) patch antenna for IEEE 802.11b, 802.11g, 802.11n and IEEE 802.16 standard-based WLAN and WiMAX bands is presented in this paper. The designed antenna consists of a trapezoidal-shaped radiating patch with several asymmetric stubs embedded at the right periphery of the trapezoid. The antenna has a miniaturized size ($50 \times 50 \times 1.6$ mm^3) and shows good CP radiation at 2.4 GHz and 3.5 GHz frequencies. A bandwidth ($S_{11} \leq -10$ dB) of 1–6.5 GHz is attained by the proposed antenna, and it shows a flat gain in the complete frequency band.

Keywords Axial ratio · Broadband · Monopole

1 Introduction

Planar monopole antennas are a preferred choice for wireless communication applications. The key strengths of the planar monopole antennas lie in their small dimensions, low-contour, wide bandwidth, omnidirectional radiation patterns and their capability of easy integration with other microwave components and circuitry. One of the major limitations of an ordinary microstrip antenna is its narrow bandwidth, which has been improved by using monopole geometry. In the last decade, a number of monopole shapes like square, circular, annular, elliptical, trapezoidal, hexagonal have been presented by various researchers worldwide [1, 2]. Many other monopole

B. J. R. Sahu (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be) University, Bhubaneswar 751030, India
e-mail: bharatjyotisahu@soa.ac.in

S. Kumar
School of Electronics Engineering, Kyungpook National University, Daegu 41566, Republic of Korea

S. Saxena
Department of Electronics Engineering, Indian Institute of Technology (ISM), Dhanbad 826004, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_26

designs with the defected ground surfaces have also been presented for bandwidth enhancement, cross-polarization suppression, achieving notch bands and polarization diversity [3, 4].

Circularly polarized (CP) antennas are the most preferred antennas for mobile handheld wireless transmission in GPS, RFID, WLAN, WiMAX and Bluetooth. CP antennas effectively reduce the multipath interference issues and are seen to be unaffected by the polarization mismatch problems. The strength of the signal remains the same, even if there is any change in the direction of the receiver, with respect to the transmitter. This provides the benefit of smooth communication between the transmitter and the receiver as compared to linearly polarized (LP) antennas. In the last few years, a number of CP antennas have been presented by many people [5, 6]. The designs presented were generally composed of trimmed corners, modified ground, integrated active devices, etc., for exciting CP current modes. A few design configurations were comprised of distinct feeding arrangements [7]; however, such antenna configurations are hard to manufacture and occupy additional space also. Small axial ratio (AR) bandwidth is one of the main limitations of the proposed CP antennas. The CP monopole antennas are used in modern handheld systems as monopole antenna is compact and exhibits broad bandwidth. But, a very few monopole antennas are proposed in the literature for CP radiation [8–10]. A coplanar printed monopole antenna was presented in [8], with multiband LP and CP radiation characteristics. In reference [9], a monopole antenna with uneven arms was proposed for CP. In [10], a monopole antenna with coplanar waveguide (CPW) feed was presented. But the major drawback of these designs is their narrow AR bandwidth.

In this communication, a small, low-contour monopole antenna with CP bands is proposed. The designed microstrip line-fed antenna consists of a trapezoidal radiating patch with several asymmetric stubs embedded at the right periphery of the trapezoid. The ground of the antenna is made up of a simple finite rectangular plane. The proposed antenna offers broad radiation characteristics with impedance bandwidth varying from 1 to 6.5 GHz. The antenna covers almost all the useful frequency bands and shows CP at two bands (2.4 and 3.5 GHz). The antenna may be a superior candidate for IEEE 802.11b, 802.11g, 802.11n standards in the 2.4 GHz band and IEEE 802.16 standard in the 3.5 GHz band. The 3-D EM tool ANSYS HFSS® is utilized for simulating and optimizing the parameters of the proposed monopole antenna.

2 Antenna Configuration

A schematic of the designed antenna is illustrated in Fig. 1. FR-4 substrate with thickness 1.6 mm and dielectric constant 4.4 is considered for the designing of the antenna structure. The antenna structure is composed of a rectangular ground plane, microstrip line feed and trapezoid shape radiator with implanted inverted L-shaped stubs. The inverted L-shaped stubs are used for the excitation of current modes of nearly the same magnitude and a 90° phase deviation for generating the CP band.

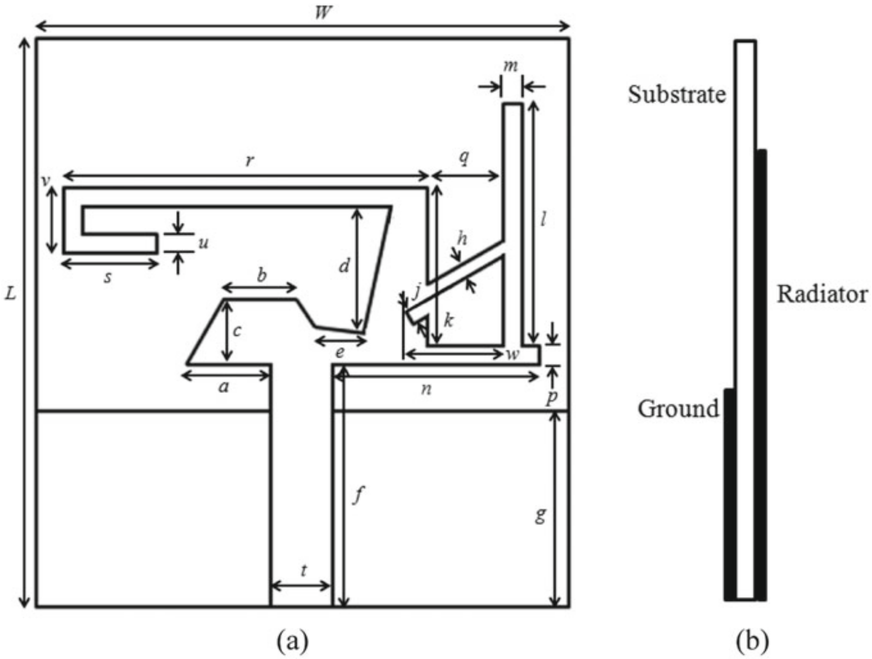


Fig. 1 Geometry of the designed monopole antenna: **a** top view and **b** side view

Table 1 Parameters of the optimized antenna (in mm)

Parameter	Value (mm)	Parameter	Value (mm)
L	50	*k*	10
W	50	*l*	28
g	15	*m*	1
f	17	*n*	23.5
t	3	*p*	1
a	8	*q*	11.8
b	5	*r*	33.2
c	6	*s*	9.5
d	8	*u*	1
e	6.1	*v*	4
h	0.5	*w*	11.8
j	0.5		

The dimensions of the arms of the L-shaped stubs are changed to vary and optimize the amount of AR bandwidth. The effective specifications of the proposed monopole structure are shown in Table 1.

3 Results and Discussion

Figure 2 illustrates the S_{11} of the designed monopole antenna. From the figure, it can be concluded that the impedance bandwidth of the proposed antenna is 146.7% (1–6.5 GHz); hence, the considered monopole antenna covers almost all the frequency bands required for contemporary wireless communication applications. In Fig. 3, the AR behavior of the proposed monopole antenna is represented. The CP bands of the antenna completely cover the 2.4 GHz (IEEE 802.11b, 802.11g and 802.11n standards) and 3.5 GHz bands (IEEE 802.16 standard), making antenna functional for WLAN and WiMAX applications. The simulated current distributions at 2.4 and 3.5 GHz CP bands are symbolized in Fig. 4a, b, respectively.

The radiation patterns of the designed monopole antenna are presented in Fig. 5. The patterns are plotted at 2.4 and 3.5 GHz, and it is examined that in both CP bands, the antenna excites left-hand circularly polarized (LHCP) waves. The peak gain and frequency curve are plotted in Fig. 6. The graph presents that the monopole antenna exhibits flat gain behavior in the complete frequency range. Furthermore, the radiation efficiency of the monopole antenna is shown in Fig. 7 and it can be concluded from the graph that the designed antenna radiates more than 80% of the incident energy.

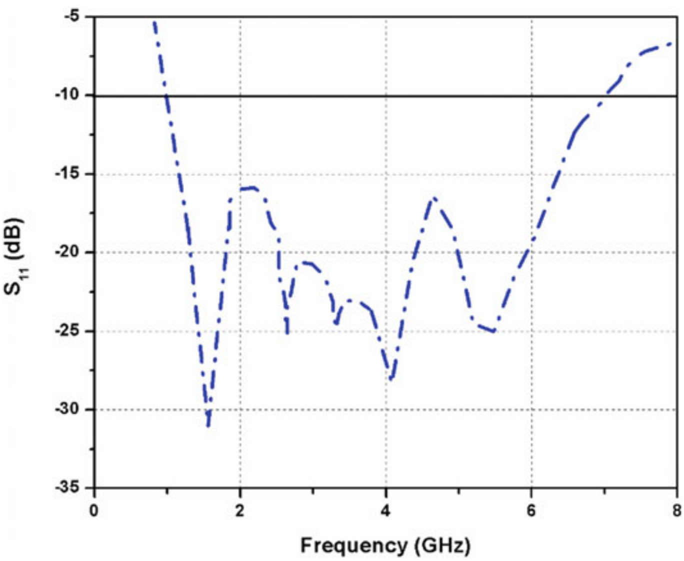


Fig. 2 S_{11} of the proposed monopole antenna

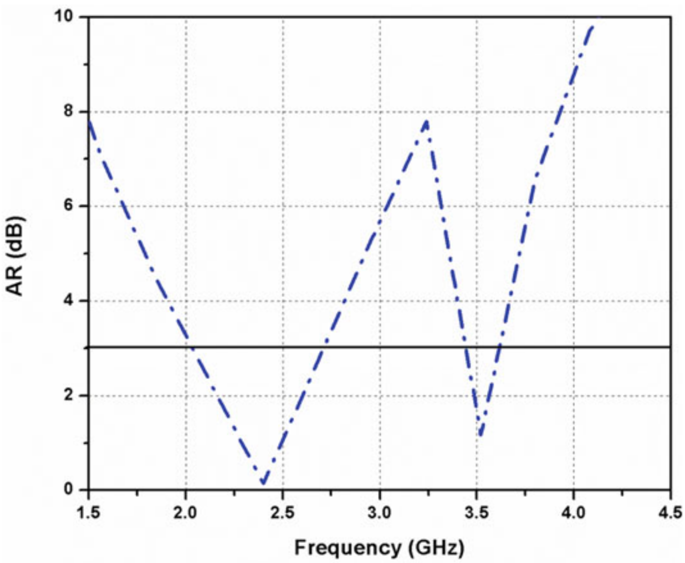


Fig. 3 AR of the proposed monopole antenna

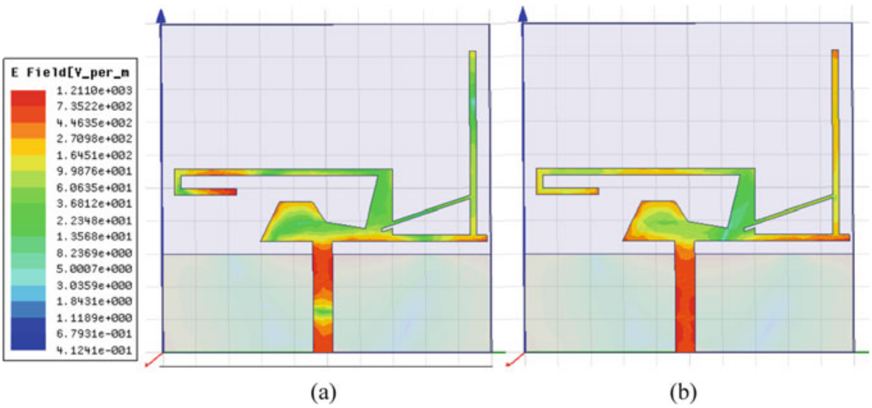


Fig. 4 Current distribution on the radiator of the proposed monopole antenna: **a** 2.4 GHz and **b** 3.5 GHz

4 Conclusion

A low-contour microstrip line-fed CP monopole antenna is demonstrated for modern wireless communication. Asymmetrical L-shaped stubs are used in the radiator for obtaining circular polarization. The unequal arms of the L-shaped stubs produce orthogonal current modes required for circular polarization excitation. It may also

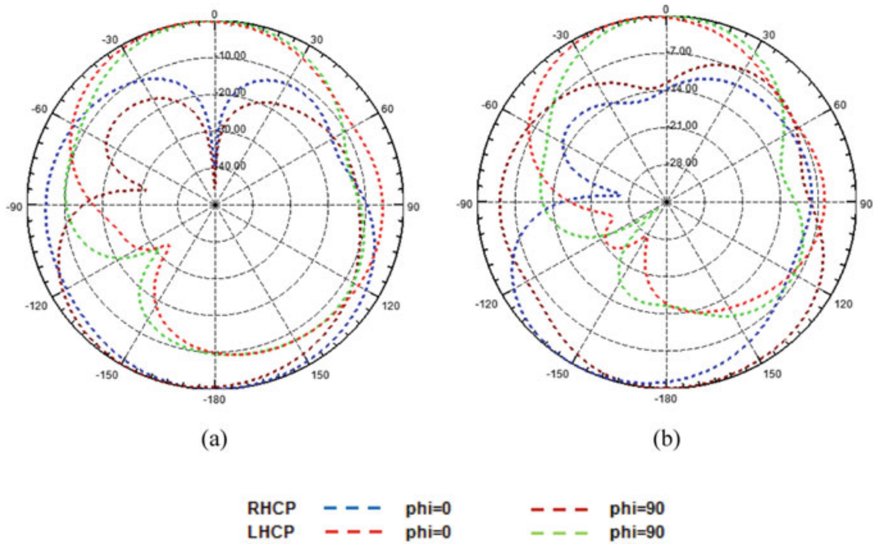


Fig. 5 Radiation plots of the designed monopole antenna

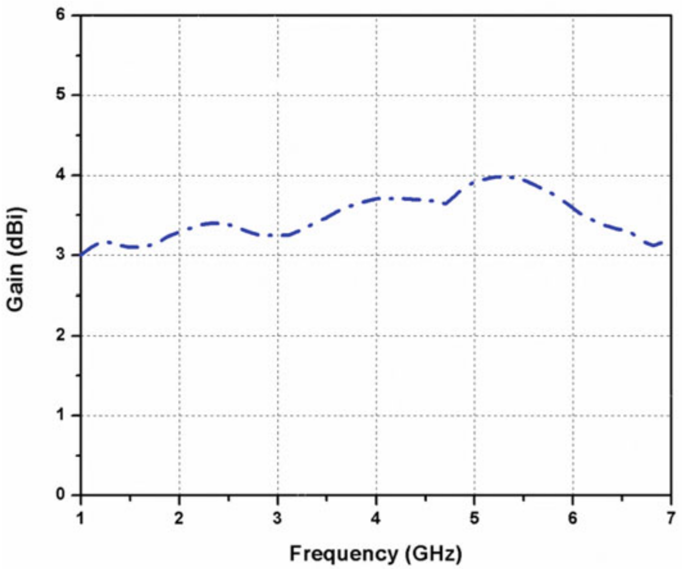


Fig. 6 Gain of the designed monopole antenna

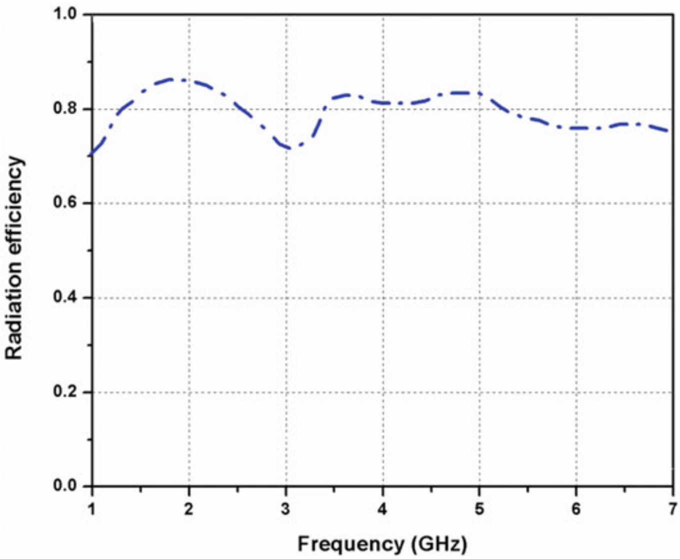


Fig. 7 Radiation efficiency of the proposed monopole antenna

be noted that good CP radiation is observed at 2.4 and 3.5 GHz bands, which makes the proposed antenna a useful candidate for WLAN and WiMAX applications.

References

1. Chen, Z.N., Ammann, M.J., Qing, X.M., Wu, X.H., See, T.S.P., Cai, A.: Planar antennas. IEEE Microw. Mag. **7**(6), 63–73 (2006)
2. Ha, J., Al-Tarifi, M.A., Filipović, D.S.: Design of wideband combined annular slot-monopole antenna. IEEE Trans. Antennas Propag. **64**(9), 4138–4143 (2016)
3. Singh, A., Singh, S.: A novel CPW-fed wideband printed monopole antenna with DGS. AEU–Int. J. Electron. Commun. **69**, 299–306 (2015)
4. Saxena, S., Kanaujia, B.K., Dwari, S., Kumar, S., Tiwari, R.: A compact microstrip fed dual polarised multiband antenna for IEEE 802.11 a/b/g/n/ac/ax applications. AEU–Int. J. Electron. Commun. **72**, 95–103 (2017)
5. Lu, J.H., Yu, H.C., Wong, K.L.: Compact circular polarization design for equilateral triangular microstrip antenna with spur lines. Electron. Lett. **34**(21), 1989–1990 (1998)
6. Tong, K.F., Wong, T.P.: Circularly polarized U-slot antenna. IEEE Trans. Antennas Propag. **55**(8), 2382–2385 (2007)
7. Liu, Q., Chen, Z.N., Liu, Y., Li, C.: Compact ultrawideband circularly polarized weakly coupled patch array antenna. IEEE Trans. Antennas Propag. **65**(4), 2129–2134 (2017)
8. Augustin, G., Denidni, T.A.: Coplanar waveguide-fed uniplanar trapezoidal antenna with linear and circular polarization. IEEE Trans. Antennas Propag. **60**(5), 2522–2526 (2012)
9. Ghobadi, A., Dehmollaian, M.: A printed circularly polarized Y-shaped monopole antenna. IEEE Antennas Wirel. Propag. Lett. **11**, 22–25 (2012)
10. Wang, C.J., Hisao, K.L.: CPW-fed monopole antenna for multiple system integration. IEEE Trans. Antennas Propag. **62**(2), 1007–1011 (2014)

Detection of XSS Vulnerabilities of Web Application Using Security Testing Approaches

Sanjukta Mohanty and Arup Abhinna Acharya

Abstract The security testing of web applications is a method which ascertains that a particular web application defends the web data and retains the functionality as anticipated. Any loopholes or improper validation or lack of sanitization in web application causes major security flaws like cross-site scripting (XSS), SQLi, etc., and the intruder avails the advantage of it which results in deformation of Web sites, disclosure of sensitive data, hijacking of cookies or session, etc. So to overcome these issues, research practitioners have suggested several security testing approaches for detection and removal of vulnerabilities in web application. In this research article, we provide several XSS vulnerability detection approaches using static analysis and evolutionary genetic algorithm (GA) but no study has ever revealed the false-negative results of source code. So, we plan to fill this research gap by integrating the approach of static taint analysis with GA. Static taint analysis can report candidate false negatives, and genetic algorithm will generate the test cases for exposing the actual vulnerabilities.

Keywords Security testing · Web application vulnerabilities · XSS · SQLi · GA · Taint analysis

1 Introduction

Web applications are the application program which run either in the client machine or in the server. But together they make the web page more interactive, convenient and versatile. Web applications provide various facilities to user through Internet. The root cause of vulnerabilities is the diverse technologies collaborated in the web, bad programming practices and errors in the programs. These errors or loopholes

S. Mohanty · A. A. Acharya (✉)
School of Computer Engineering, KIIT University, Bhubaneswar 751024, India
e-mail: aacharyafcs@kiit.ac.in

S. Mohanty
e-mail: mailtorani.sanjukta@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_27

or bugs that exist on the web application can be exploited by intruders known as web vulnerabilities. Almost all web applications handle the user's sensitive data and carry out the different activities like business-critical activity, online taxpaying, banking, online shopping, social media accounts, etc. Most of the social communication, business and financial transaction carried out by the user are dependent on web application. However, the web vulnerabilities impose a restriction on the user activities including redirection of user to malicious site, theft of personal credentials through cookies and session, installment of malicious and other illegal activities [1]. In order to overcome these issues, the security programmers are using various techniques. Hence, the process of security testing is highly demanding as it determines whether a web application defends data and retains functionality as anticipated [2]. According to (OWASP-2017) [3], the most dangerous vulnerabilities of web system are injection (SQLi) and XSS vulnerabilities. XSS allows an intruder to accomplish malicious scripts on the web browser of victim consequencing the various side effects like deformation of Web site, loss of data and information, theft of cookies, etc. The various techniques or the solutions suggested for addressing the issues of XSS are static analysis, dynamic analysis, secure programming, modeling and others [4]. In this research work, we are focusing on detection of XSS vulnerabilities by adopting different approaches of static analysis.

The structure of the research work is planned as follows: Section 2 presents some background details of security vulnerabilities of web application. Related works are discussed in Sect. 3. Discussion is presented in Sect. 4, and Sect. 5 contains conclusion of the research work and suggests some future work.

2 Background

Web application vulnerabilities are the bugs or flaws in application which permits the intruders to do something malicious. As web application is widely used and open to all, these applications are more prone to security vulnerabilities than other applications. The primary reason for their vulnerability is due to the improper input validation or lack of sanitization implemented in the application. Security testing of web system is a process which ensures whether the web applications are free from any kind of vulnerabilities. To address the issues of XSS, security testing approaches are divided into five types: static, dynamic, secure programming, modeling and others [5]. In static security testing, the source code is analyzed to find the security bugs. In this process, the data and control flow path are modeled to find the strength and weakness of security. Many studies have proposed the different static analysis techniques for detecting XSS vulnerabilities like taint analysis, genetic algorithms, symbolic code execution, string analysis, precise alias analysis, etc. In dynamic analysis, the SUT is executed on real data. It is a type of black box testing [5]. In this type of testing, vulnerabilities are detected during execution time. Malicious scripts are sent to the SUT, and responses are evaluated. In this case testing, engineer perceives the application from the attacker's point of view. So, dynamic analysis

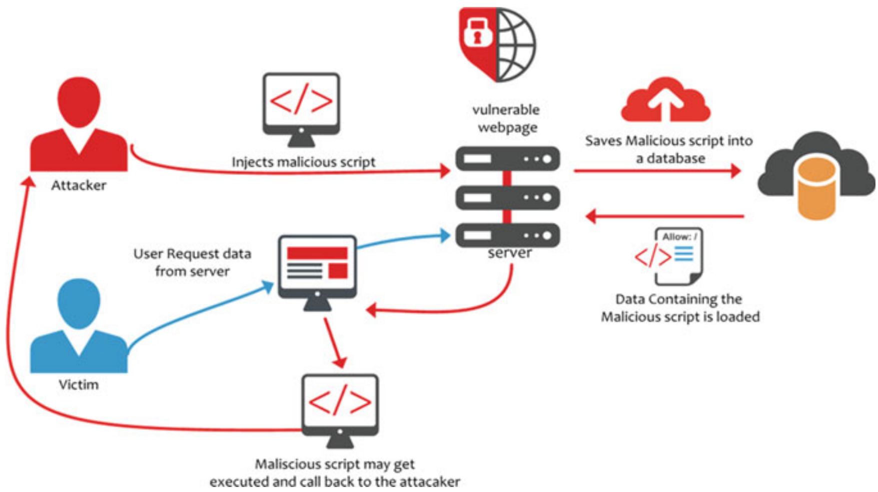


Fig. 1 Flow of XSS attack [6]

is more actionable than static analysis in vulnerability detection [2]. We focus on both the static and dynamic analysis techniques. We can review the source code by static analysis techniques without running the application. However, to detect the vulnerability executable codes are required in dynamic analysis approaches.

We focus on the two major vulnerabilities like XSS and SQLi of web application as these two are the most dangerous [3]. XSS vulnerability occurs due to the improper validation of user inputs. The malicious scripts or untrusted data is executed in the victim's browsers. There are three types of XSS vulnerability: reflected, stored and DOM-based XSS vulnerabilities. Reflected XSS vulnerabilities executed at user's web browser when the user provides input to the browser, e.g., commonly occurred error and greeting messages. In case of stored XSS vulnerability, the scripts of the user are stored in the database server. Later on, these stored scripts are retrieved and used in the response page, e.g., forums, blogs, social networking sites, etc [2]. In the DOM-based attack, the script resides in the client side and works over the Document Object Model rather than HTML code snippet. Figure 1 shows how the scenario of XSS injection flaws occurs. OWASP [3] defines SQLi as an injection attack through which an intruder changes the structure of the original queries by injecting SQL code in the input fields of web application form to get unauthorized access of database.

3 Related Work

To detect the security vulnerability, research practitioners have suggested a variety of techniques. Each approach has some objective to satisfy, and we have discussed some of these methodologies sequentially.

Jovanovie et al. [7] discovered the vulnerable points of the program by using the techniques like flow-sensitive, inter-procedural and context-sensitive data flow analysis statically. They also employed other techniques like alias analysis and literal analysis to further improve the correctness and precision of the result. For detecting the taint style vulnerabilities, they used open-source prototype tool: Pixy. Main objective of this tool is to detect the XSS vulnerabilities in PHP scripting. Limitation of their research was not supporting the object-oriented features of PHP. G. Wassermann [8] presented a combined approach of taint analysis with string analysis. Their analysis could not detect DOM-based XSS. Kie˙zun [9] proposed an attack creation technique for cross-site scripting (XSS) and SQL injection which produce a set of test data. They also used a tool (Ardilla) to detect vulnerabilities with high accuracy. The most dangerous stored (second-order) XSS attack can be accurately and effectively detected by Ardilla tool. The generated test data is used to execute the program, and the proposed approach checks the data flows. If data flows from source to sink, then the data is modified with the help of a library of attack patter and tested again for the vulnerabilities. Jovanovic [10] used static source code analysis approach to identify the issues of vulnerable web application. To find the more accuracy of the result, they used the two static analysis approaches like taint analysis and precise alias analysis. To detect the SQL and XSS vulnerabilities, they have used the Pixy tool. Avancini et al. [11] detected only the reflected XSS type of vulnerability. They integrate the approach of static taint analysis and genetic algorithm into their proposed approach. Static analysis identifies potential vulnerable statements in web source code, and genetic algorithm generates input values which are enforced to traverse the vulnerable control flow graph (CFG). Only the feasible paths are covered by the test cases. Their approach was effective in creating the security test cases for reflected type of cross-site scripting vulnerabilities. Persistent and DOM-based XSS vulnerabilities are not implemented. Agosta [12] presented a methodology and tool known as Codeminer, for identification of vulnerabilities statically. It accurately tracks the string parameters passed to a sink statement. This tool was much faster and effective than the Pixy tool. Avancini et al. [1] combined the approach of GA with symbolic code execution for generating the security test cases which cover more vulnerabilities in less time, hence increasing the performance. Moataz et al. [12] enhanced the approach adopted by Avancini and Ceccato [8]. They presented a test data generator which is GA-based. This test data generator creates multiple test data to cover multiple target paths of XSS vulnerabilities. However, the experiments they performed are able to find out stored and reflected XSS vulnerabilities. It does not expose DOM-based XSS vulnerabilities. Furthermore, their approach has some limitation like they were unable to remove the infeasible paths from the program execution. Avancini [13] presented an approach to generate the test cases automatically which can be able to detect the XSS vulnerabilities. Their approach used taint analysis, genetic algorithm and constraint solver together to find the XSS vulnerabilities. They only detected the reflected type of XSS and are unable to generate test cases for finding the false-positive and false-negative results. Marashdih et al. [18] proposed an enhanced approach of Moataz [12] in PHP web application using taint analysis and genetic algorithm. They mainly focused on eliminating the infeasible path from control flow graph (CFG) to reduce the false-positive rate in the result.

Although there are many approaches been proposed for detecting the XSS vulnerability, still it continues to persist in the web applications. The main objective of this article is to reduce the false-positive and false-negative rates by determining the actual vulnerable paths.

4 Discussion

To detect the XSS vulnerabilities of web applications, we have discussed some security testing approaches, tools and languages, limitations and their area of focus in Table 1. Most of the research articles have discussed static taint analysis and genetic algorithm in PHP and Java web applications. Their approaches can be debated because they also need to remove the false-negative results from the source code. Although Marashdih [14] removed the infeasible paths than applied GA to reduce the false-positive rate from the program, still the false-negative results are the issues. Also, they have only detected the XSS vulnerabilities and other vulnerabilities like SQLi and CSRF are still needed to be identified by GA. Therefore, we want to use the approach of GA with static taint analysis which can remove not only the infeasible path but also the false-positive as well as false-negative results of the source code and it is depicted in Fig. 2. Here, we have presented a new security framework based on the approach Moataz et al. [2] which is diagrammatically represented in Fig. 2, where the source code is statically analyzed and a CFG is drawn which acts as an input to the GA. GA will also take another input from the database of XSS and SQLi attack pattern and generates the test data which will traverse the CFG to find out the more vulnerable paths and also count the number of false-positive and false-negatives results.

5 Conclusion and Future Work

Cross-site scripting and SQLi injection vulnerabilities are the major security holes in web applications which can lead to hijacking of user accounts and cookies, deformation of Web sites, loss of data in database, etc. Though several research studies have been performed pertaining to the issues of XSS vulnerabilities, still their results seem to be inefficient. Static taint analysis still contains the false-positive and false-negative results which need to be lessened. Also, the dynamic analysis techniques

Table 1 Summary of the literature review

Sl. no.	Articles	Security testing approaches	Tools and languages	Results
1	Jovanovie et al. [7], 2006	Static analysis approaches like flow-sensitive, inter-procedural, and context-sensitive data flow analysis	Pixy and ASP	Provide lower degree of precision in detecting the XSS vulnerabilities
2	Wassermann et al. [8], 2008	Static analysis approaches like taint analysis and string analysis	Nil	This static analysis could not detect DOM-based XSS
3	Kie˙zun et al. [9], 2009	Dynamic taint analysis and mutation of input	Ardilla	This approach detected only the stored XSS vulnerabilities and SQL injection vulnerabilities
4	Jovanovic et al. [10], 2009	Static analysis approaches	Pixy and PHP	XSS and SQLi vulnerabilities detected. Provide a higher degree of precision in detecting the vulnerabilities. Rate of false positive is low
5	Avancini et al. [11], 2010	Static taint analysis and genetic algorithm	Pixy and PHP	This approach detects too many false positives. Only reflected XSS vulnerability is detected. Stored and DOM-based vulnerabilities are not detected
6	Avancini et al. [13], 2011	Static taint analysis, GA and constraint solver	Pixy	Only reflected XSS vulnerability is detected. No test cases are provided for determining false-positive and false-negative results, also no criteria found in removing infeasible paths

(continued)

Table 1 (continued)

Sl. no.	Articles	Security testing approaches	Tools and languages	Results
7	Agosta G et al. [12], 2012	Static analysis	Codeminer	Identified the XSS and SQL injection vulnerabilities
8	Avancini et al. [1], 2013	GA and symbolic code execution	Pixy and PHP	Focused on only reflected XSS vulnerabilities
9	Hydara et al. [15], 2015	Genetic algorithm	Java	Focused on DOM-based XSS vulnerability More false-positive rate
10	Moataz A et al. [2], 2016	Static analysis and genetic algorithm	Pixy and PHP	DOM-based XSS vulnerabilities are not covered
11	Marashdih A.W et al. [14] 2017	Static analysis and genetic algorithm	Pixy and PHP	Reflected, stored and DOM-based vulnerabilities are covered Infeasible paths are discarded Low false-positive rates

required to be improved to get the high accuracy. As genetic algorithm has succeeded in detecting the XSS vulnerability, we can improve the fitness function and GA operators to generate minimum number of security test cases to identify more vulnerable paths in the CFG so as to remove the false-positive and false-negative results from XSS and SQLi injected web applications.

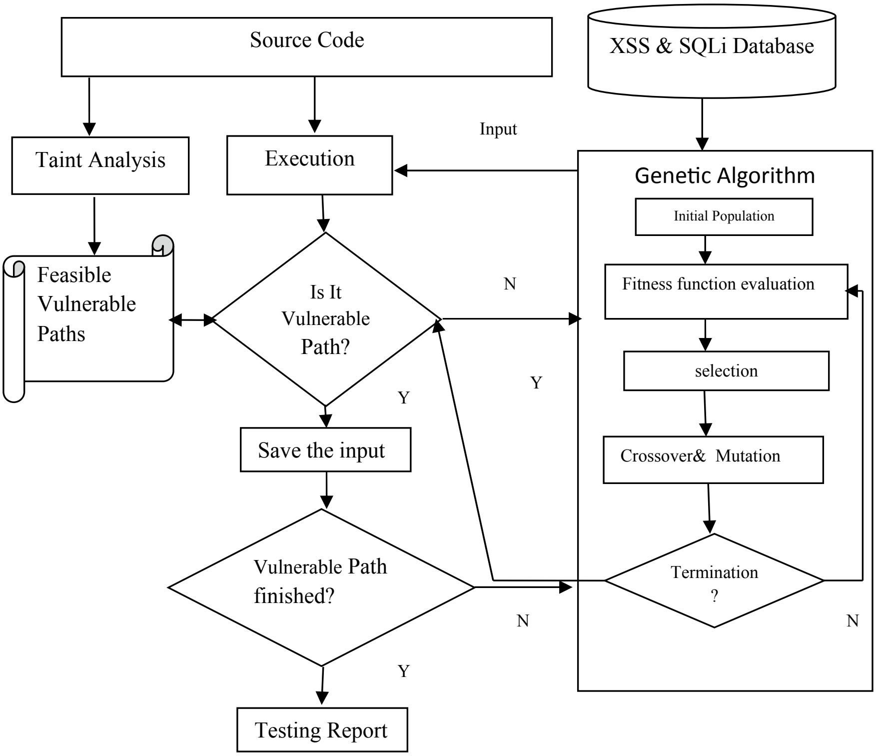


Fig. 2 Proposed approach [2]

References

1. Andrea, A., Mariano, C.: Comparison and integration of genetic algorithms and dynamic symbolic execution for security testing of cross-site scripting vulnerabilities. Inf. Softw. Technol. **55**(12), 2209–2222 (2013)
2. Moataz, A.A., Fakhreldin, A.: Multiple-path testing for cross site scripting using genetic algorithms. J. Syst. Architect. **64**, 50–62 (2016)
3. OWASP, top-10 threats for web application security, Available. https://www.owaspp.org/index.php/Top_10_2017. (2017)
4. Hydara, I., Abu, B.M.S., Hazura, Z., Novia, A.: Current state of research on cross-site scripting (XSS)–A systematic literature review. Inf. Softw. Technol. **58**, 170–186 (2015)
5. Felderer, M., Matthias, B., Martin, J., Achim, D.B., Ruth, B., Alexander, P.: Security testing: a survey. In: Advances in Computers, vol. 101. Elsevier, Amsterdam, pp. 1–51 (2016)
6. https://secure.wphackedhelp.com/blog/wordpress-xss-attack/. Last Accessed 2019
7. Zhang, X, Zhi-Jian, W.: A static analysis tool for detecting web application injection vulnerabilities for ASP program. In 2010 2nd International Conference on E-business and Information System Security. (2010)
8. Wassermann, G., Zhendong, S.: Static detection of cross-site scripting vulnerabilities. In: ACM/IEEE 30th International Conference on Software Engineering, pp. 171–180. IEEE, (2008)
9. Ernst, M.: Automatic Creation of SQL Injection and Cross-Site Scripting Attacks. (2008)

10. Jovanovic, N., Christopher, K., Engin, K.: Static analysis for detecting taint-style vulnerabilities in web applications. J. Comput. Secur. **18**(5), 861–907 (2010)
11. Avancini, A., Mariano, C.: Towards security testing with taint analysis and genetic algorithms. In: Proceedings of ICSE Workshop on Software Engineering for Secure Systems. ACM, pp. 65–71 (2010)
12. Agosta, G., Alessandro, B., Antonio P., Gerardo, P.: Automated security analysis of dynamic web applications through symbolic code execution. In: 9th International Conference on Information Technology-New Generations. IEEE, pp. 189–194 (2012)
13. Avancini, A., Mariano, C.: Security testing of web applications: a search-based approach for cross-site scripting vulnerabilities. In: IEEE 11th International Working Conference on Source Code Analysis and Manipulation. IEEE, pp. 85–94 (2011)
14. Marashdih, A.W., Zarul, F.Z., Herman, K.O.: Web security: detection of cross site scripting in PHP web application using genetic algorithm. Int. J. Adv. Comput. Sci. Appl. **8**(5) (2017)
15. Hydara, I., Sultan, A.B.M., Zulzalil, H., Admodisastro, N.: Cross-site scripting detection based on an enhanced genetic algorithm. Indian J. Sci. Technol. **8**(30), 1–7 (2015)

Artificial Intelligence (AI)

Evolutionary Hybrid Feature Selection for Cancer Diagnosis

Bichitrananda Patra, Lambodar Jena, Santosini Bhutia, and Soumen Nayak

Abstract How to picking a lesser subsection out of the thousands of genetic factors in microarray statistics is vital for the perfect classification of phenotypes. Extensively recycled approaches naturally rank genes allowing their discrepancy terminologies among phenotypes and preference the topmost categorized genes. While microarrays can extent the ranks of thousands of genes per sample, situation mechanism microarray studies regularly include no more than numerous dozen samples. Normal classifiers do not effort fine in these circumstances where the number of features distant surpasses the amount of illustrations. Choosing only the features that are most pertinent for discerning between the two types can help construct better classifiers, in terms of both accuracy and efficiency. We detect that feature sets so gained have certain redundancy and study methods to minimize it. In this paper, it is suggested that the least severance and extreme significance feature selection framework. Here there are two general approaches of feature subset selection, more specifically, wrapper and filter methods and then twisted a novel classical called hybrid model by merging the physiognomies of the two stated simulations for gene selection. Elephant search (ES) based optimization is planned to choose finest gene terms commencing the bulk of microarray data. We have similarly equated the gene collection performance of the filter model, wrapper model, and hybrid model. This leads to significantly improved class forecasts in general experiments on eight gene expression data sets. Enhancements are experimental reliably among three convolution neural network classification methods.

B. Patra (✉) · L. Jena · S. Bhutia · S. Nayak
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: bichitranandapatra@soa.ac.in

L. Jena
e-mail: lambodarjena@soa.ac.in

S. Bhutia
e-mail: santosini.bhutia@gmail.com

S. Nayak
e-mail: soumennayak@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_28

Keywords Evolutionary algorithm · Fuzzy rough set · LVQ · SOM · BP

1 Introduction

Classifying different substances deposited through a microarray preparing dataset has become an acute job now several practical uses of data mining and pattern recognition [1, 2]. The comparative content through mRNA consistent toward genes can be observed in gene expression profiles. Since the results of the discriminate analysis of the dataset characterize the state of the cell at the molecular layer has the potential of being used by a medical investigative tool. Therefore, the objective of the classification of microarray datasets is to design an efficient model to discriminate the variability of expressed genes and may be used for predicting unknown samples. The inadequate quantity of sections in evaluation in height dimensional of the dataset and the investigational differences in gene appearance intensities become a major challenge in microarray classification.

The selection of efficient feature extraction techniques and predictive models provide high classification accuracy for the microarray dataset. As such, the investigations show that a minor quantity of gene expression data has a solid association with certain phenotypes associated with the entire quantity of genes available. Consequently, selections of differentially expressed relevant predictor genes correctly analyze gene expression profiles and also play a crucial role in the classification process. The subset of potential genes identified by feature selection technique correctly distinguishes the model periods. Consequently, noble selection technique intended for genes, applicable for model grouping based on the quantity of genes examined is desired to escalation the analytical correctness.

Regularly, expression levels of the genes in the microarray dataset are expressed in the form of a matrix, where each column corresponds to the expression level of one gene and each row to one sample. Every cell in this environment triggers countenances close ailment. It denotes the actual assessment. Classically datasets comprise although quantity situations as assortments starting 10–100. To manage through this tremendously high dimensionality, old-fashioned machine learning approaches that is linear discriminate analysis [3] and nearest- neighbor [4] and additional approaches fuzzy logic [5] and NN [6] cannot be working efficiently and competently.

Some of the furthermost effective relations of data fall approaches have been invented by evolutionary calculation [7, 8]. Evolutionary algorithm has been used to dissimilar data reduction problems, demonstrating them as combinatorial solutions [9]. Assessment of every one of the population is established on its appropriateness rate computed from fitness purpose, i.e., further tough the result, advanced is the chance to replicate. This development is repeated until the ending ailment is fulfilled. As a result, Genetic Algorithm examines all practicable results which progresses the chance of searching uncharted provinces and the opportunity of realizing a complete best/near optimum result.

Fuzzy set [10] and Rough set [11] discourse dual significant, matching features of faulty data and information. [12, 13] A hybrid fuzzy rough set exemplary was major projected and advanced extended by several journalists, being practical effectively in several fields [14]. Feasibly, the best prominent competence of these additions is that they allow experts to relate rough sets investigation completed the data sets comprising uninterrupted data, conflicting to pure rough set approaches, which may not be used over incessant data set short of discretizing them at a preceding phase.

Second portion is considerable examination deprived of preceding expectations around the measurements of contribution data and has been recycled in many feature identification jobs with pattern recognition [15], image dissection, and biometrics model [16]. In specific, they have a tough classification and learning control to signify inherent awareness of the assumed data.

The chapter contains, a FRESNN hybrid mode for feature selection through coalescing with evolutionary elephant search (ES) and fuzzy rough (FR) sets is projected and the usefulness of the selection technique is weighed by different classifiers. The data reduction is executed by fuzzy rough (FR) and elephant search (ES) algorithms in dissimilar methods, filter, and wrapper [17]. By the use of machine learning Nearest Neighbor's algorithm to calculate ES solution in wrapper search (ESNN) techniques and filter search (FRES) calculate the core deduct set computing the feature each effective deduct set. Each contestant solution is calculated built on the situation of fitness by selection, crossover, and mutation. A selection of entities is recycled to search the overall best features.

Here, we have compared the classification correctness through different module neural network algorithms, i.e., LVQ, SOM, and BP for filter, wrapper, and hybrid feature selection techniques. The planned technique is established with six microarray standard data sets from which three are binary and other three are multi-class to launch the objective of the effort.

The organization of the chapter is as follows: a summary of the learning structure is specified in Sect. 1. The proposed learning scheme which is a combination of the evolutionary hybrid search method is described in Sect. 2. Section 3 presents the experimental results and discussion that explain the performance of the proposed scheme. Section 4 concludes the future scope.

2 Related Work

The major important component of FRESNN classical is fuzzy rough feature deduction techniques that calculate the deduct set. The additional important component of FRESNN classical is its search technique for choosing best subsections of features. To achieve optimized results, we have selected elephant search (ES) that evolutionary procedure for complete the searching. Elephant Search (ES) [18] is an exceedingly nonlinear, multimodal universal optimization method stimulated by the biotic habits of elephant herds. The Elephant search is measured to be an upright optimization method for its strengthening in limited search space for improved clarification. It delivers inclusive optimum keys through wrapper realistic search space deprived of dropping into limited goals.

The outline specifics of the Elephant search (ES) is as follows:

- Throughout dry climate, deficiency of water forms an irresistible badly behaved for every animal, particularly for enormous nature's like elephant. In the nastiness of the callous alive situations formed throughout the efficiency, the elephant is fortified sound to live for discovery water resources. Elephants can exploit once or new communiqué arrangements to hunt the water assets, contingent at his existing situations.
- Naturally, they travel essential for invention foods and resources of water. If the dearth area dimension is trivial, then they do not go generally far away. But the dearth area is outsized; they can travel to hyperfocal distance in hunt of water, where elephants can visit till the raining period.
- Elephants too support for other creatures by generating or else discovery different aquatic sources. They can illustration generous and communal conduct through opposing state alike dearth.

3 Experimental Results and Discussion

At here we have used 6 microarray datasets to ascertain the rationality of the projected structure. On the view of 6 datasets, 3 datasets, i.e., NIH, Stanford, and tumor are binary categories and the other 3 are multi-class, i.e., Lung, Brain, Tumor, SRBCT. We have taken DLBCL and GEMS data samples from three dissimilar bases. The records exploited in this experimentation part are concise in Table 1. By WEKA machine learning platform we have done our experiment for exertion.

In this work, performances of the classification algorithms are measured using predictive accuracy:

$$\mathrm{Acc} = \mathrm{TP} + \mathrm{TN}/(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}) \tag{1}$$

TP True Positive, i.e., correct classifications of positive samples.
TN True Negative, i.e., correct classifications of negative samples.

Table 1 Experimented Dataset

Dataset	# of genes	Trained models	Tested models	Authors
NIH	7399	160 (72:88)	80 (30:50)	Rosenwald et al. [17]
Stanford	4026	47 (24:23)	0	Alizadeh et al. [2]
Tumor	7129	77 (58:19)	0	Shipp et al. [13]
Lung	12,600	122 (58:21:20:6:17)	81 (81:0:0:0:0)	Bhattacharjee et al. [3]
Brain tumor	5920	90 (60:10:10:4:6)	0	Pomeroy et al. [18]
SRBCT	2308	83 (29:11:18:25)	0	Khan et al. [11]

FP False Positive, i.e., incorrect classifications of positive samples.
FN False Negative, i.e., incorrect classifications of negative samples.

We have compared our results with published methods based on the same dataset and observed our technique is analogous to others to a certain level. Here outcome stands have sustained the leaning of accuracy, generally what we suppose as of the perception of filter, wrapper, and hybrid model in [19].

From six trained datasets, we observe that all classifiers attain the finest outcomes while they effort on particular datasets. However, there is no generalized approach designed for tumor classifications problem so far.

Aimed at total module adaptive NN classifiers, fix set the parameters of backpropagation algorithm as 30 layers at hidden layer 2 output. We considered 0.01–0.50 as the learning rate, momentum 0.9, and 500 number of maximum iterations. Similarly, we had taken for LVQ classifier as 25 codebook vectors, range from 0.1 to 0.50 learning rate, static as learning function, 0.1 momentum and 1000 as maximum iterations. We had taken for SOM classifier as rectangular 5×5 initial, 16 neighborhood sizes. Learning rate range from 0.1 to 0.9 and maximum iterations is fixed to 2000 (Table 2).

At here we dedicated mainly on dual-class problematic. Since maximum of the complications can be condensed to dual-class, i.e., unhealthy versus regular, existence versus deadly, two contradictory subtypes of particular illnesses, and so on.

We display the classification accurateness in Table 3 of LVQ, SOM, and BP on apply feature selection gained from FRES filter selection, ESNN wrapper selection, and FRESNN hybrid selection results of six different datasets. From these six different datasets first three are binary class whereas the other three belong to multi-class problematic.

Here applied feature selection decrease prototypical, projecting accurateness from SOM (Self Organization Map) classifier are not so substantial accurateness found as former two fundamental classifiers. Identical presentation of SOM can be experimental for all six datasets. Minimum and maximum accurateness of SOM classifier is 56.63% and 79.50%, respectively. In this result, we also observed that the accuracy of filter model is least than the wrapper and hybrid model.

Table 2 Assortment of Attributes obtained from FRES, ESNN, and FRESNN model

Dataset	# of original attributes	# of attributes after filtering FRES	Selection of attributes after using ESNN	Selection of attributes after using hybrid FRESNN	Percentage of attributes selected by FRESNN
NIH	7399	304	3140	129	1.74
Stanford	4026	299	1216	132	3.27
Tumor	7129	202	2407	83	1.16
Lung	12,600	234	3313	121	0.96
Brain tumor	5920	270	2218	110	1.85
SRBCT	2308	80	976	19	0.82

Table 3 Finest assessed accurateness outcomes by three elementary NN classifiers

Datasets	FRES			ESNN			FRESNN		
	LVQ	SOM	Backpropagation	LVQ	SOM	Backpropagation	LVQ	SOM	Backpropagation
NIH	63.75	62.5	64.59	64.38	63.12	66.25	64.38	64.38	68.12
Stanford	80.85	59.58	80.85	87.23	78.72	89.37	91.49	78.72	91.49
Tumor	68.83	57.14	74.03	92.21	66.23	81.82	94.81	72.73	81.82
Lung	86.70	77.83	68.48	89.34	79.51	80.33	90.16	79.51	81.15
Brain_Tumor	74.44	66.67	68.89	78.89	68.89	70.00	83.33	71.11	72.22
SRBCT	74.70	56.63	87.96	86.75	62.65	90.36	90.36	62.65	93.98

Correspondingly, examining for analytical precision of LVQ NN in Table 3, observed that the least 63.75% is obtained whereas DLBCL-NIH, uppermost 94.81% by DLBCL-tumor (dual-class). Smallest and uppermost results are coming from FRES filter model and FRESNN hybrid model respectively.

Additional study of the outcome of module Back Propagation (BP) NN in Table 3 shows the smallest calculation accuracy of 64.59% is certain by NIH (dual-class) for RFES filter. BP recorded extreme classification accurateness of 93.98% for SRBCT (multi-class) in FRESNN hybrid model.

3.1 Analysis

Feature assortment shows a vital part in improving the classification accurateness of the problematic. Possible features affect the analytical ability of the classification model. Hence, an efficient feature selection method needs to be employed to select optimal features for the problem which results in increasing the classification accuracy. Feature selection with elephant search hybrid technique and neural network classifiers are presented for classifying gene expression datasets. The proposed technique achieves improved or is at par in comparison to the results published recently. We found the discrimination ability of the classifiers is not so significant when search strategy like filter and wrapper methods are concerned. Further, we used the same features for predicting the output of component classifiers such as BP, SOM, and LVQ and initiate the outcomes that don't improve the planned one. So, the substantial enhancements for the projected outline may be accredited to the element. The hybrid feature collection model classifies diverse mixtures of genes which improves the accurateness of classification.

In this problematic, we highlighted on feature selection structures of the problematic. On the other hand, we didn't consider the status of distinct gene and relation among them which explicitly help to become much evidence almost the disease. Above phase of the problem is left for our future research.

The only shortcoming of this method is its high computational complexity which increases the cost of the problem. However, the remarkable achievement in classification accuracy can compromise with the cost that happened due to wrong diagnosis or prognosis of the disease.

4 Conclusion

Investigational outcomes display, the evolutionary hybrid feature selection technique (FRESNN) efficiently designated subclass of marker genes for every section learner and augmented the inequitable power of the neural network classifier. Identification of potential genes from microarray dataset for classification problems has been a challenging task for the researchers. We believe our suggested method has addressed the challenge to a great extent.

At here, we introduced a hybrid FRESNN feature assortment technique for the feature selection of gene expression data. Furthermore, we recycled the classification outcome of three unitary network grounded on all the features and the result was inferior to projected technique. Hence, we trust such outstanding presentation enhancements of our technique are due to the fact that our feature selection mechanism encouraged more valuable evidence for discrimination, and the hybrid feature selection outline enhanced the constancy, robustness, and simplification of values.

References

1. Alpaydin, E.: Introduction to Machine Learning, 2nd edn. The MIT Press, Cambridge (2010)
2. Witten, I.H., Frank, E., Hall, M.A.: Data mining: practical machine learning tools and techniques. In: Morgan Kaufmann Seriesin Data Management Systems, 3rd edn. Morgan Kaufmann, Menlo Park (2011)
3. Xiong, M., Li, W., Zhao, J., Jin, L., Boerwinkle, E.: Feature (gene) selection in gene expression-based tumor classification. Mol. Genet. Metab. **73**, 239–247 (2001)
4. Dudoit, S., Fridlyand, J., Speed, T.P.: Comparison of discrimination methods for the classification of tumors using gene expression data. J. Am. Stat. Assoc. **97**(457), 77–87 (2002)
5. Ohno-Machado, L., Vinterbo, S., Weber, G.: Classification of gene expression data using fuzzy logic. J. Intell. Fuzzy Syst. **12**, 19–24 (2002)
6. Khan, J., Wei, J.S., Ringner, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson, C., Meltzer, P.S.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med. **7**(6), 673–679 (2001)
7. Eiben, A.E., Smith, J.E.: Introduction to evolutionary computing. In: Natural Computing. Springer, Berlin (2003)
8. Ghosh, A., Jain, L.C.: Evolutionary Computation in Data Mining. Springer, Berlin (2005)
9. Pappa, G.L., Freitas, A.A.: Automating the design of data mining algorithms. In: Natural Computing. Springer, Berlin (2009)
10. Zadeh, L.A.: Fuzzy sets. Inf. Control. **8**(3), 338–353 (1962)
11. Pawlak, Z.: Rough sets. Int. J. Inform. Comput. Sci. **11**, 341–356 (1982)
12. Dash, S., Patra, B.N.: Redundant gene selection based on genetic and quick-reduct algorithms. Int. J. Data Min. Intell. Inf. Technol. Appl. **3**(2), 1–9 (2013)
13. Dash, S., Patra, B.: Rough set aided gene selection for cancer classification. In: 2012 7th International Conference on Computing and Convergence Technology (ICCCT) IEEE. (2012), pp. 290–294
14. Zhai, J.: Fuzzy decision tree based on fuzzy-rough technique. Soft. Comput. **15**(6), 1087–1096 (2011)
15. Karayiannis, N., Xiong, Y.: Training reformulated radial basis function neural networks capable of identifying uncertainty in data classification. IEEE Trans. Neural Netw. **17**, 1222–1234 (2006)
16. Lee, H., Hong, S., Kim, E.: A new genetic feature selection with neural network ensemble. Int. J. Comput. Math. **86**(7), 1105–1117 (2009)
17. Jensen, R., Shen, Q.: Semantics-preserving dimensionality reduction: rough and fuzzy-rough based approaches, IEEE Trans. Knowl. Data Eng. **16**(12), 1457–147 1(2004)

18. Deb, S., Fong, S., Tian, Z.: Elephant search algorithm for optimization problems. In: The Tenth International Conference on Digital Information Management. IEEE, pp. 249–25 (2015)
19. Huei, Y.C., Li, C., Hong, Y.C.: IG-GA: a hybrid filter/wrapper method for feature selection of microarray data. J. Med. Biol. Eng. **30**(1), 23–28 (2009)

Predicting the Price of Gold: A CSPNN-DE Model

Samuka Mohanty and Rajashree Dash

Abstract Predicting the price of gold is always an attractive area for researchers and predictors willing to define its upcoming value in a more accurate and efficient way. In this study, a Chebyshev Polynomial Neural Network (CSPNN) based predicting model is suggested for the prediction of gold price. Evolutionary approaches like Particle Swarm Optimization (PSO) and Differential Evolution (DE) are used in training of the CSPNN to derive optimally tuned weights of the network. The efficiency of CSPNN model is evaluated by means of different error measures on UK/USD and MOS/USD gold price datasets. The conclusion analyzes the results to suggest a better prediction.

Keywords Gold price prediction · CSPNN · Evolutionary algorithms · PSO · DE

1 Introduction

Since a long-standing time, gold is defined as the most valuable traditional metal. It has been used as a global coinage, a possession, or simply as an object of dignity. Gold is assumed as a leading asset in financial and commercial business. As the price of gold changes within confined boundaries, it decreases the effect of hikes on it and so it is a lucrative property favoured by many stakeholders. Hence, predicting the gold price is becoming more important for investors. There is a great need for a more authenticate model of forecasting so as to define the correct price of gold based upon the changes in the gold prices in a previous time frame. Gold price prediction is already full of different time series based computational models. Artificial neural networks (ANN) have been extensively used in more recent years as the most accurate predictor model compared to traditional time series models [1–3].

S. Mohanty (✉) · R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, India
e-mail: mohantysamuka@gmail.com

R. Dash
e-mail: rajashree_dash@yahoo.co.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_29

Many ANN-based forecasters have been proposed for gold price forecasting like a GARCH-model, one using backpropagation with extreme learning machines and so on [4–9]. Recently, different Functional Link Artificial Neural Networks(FLANN) is gaining consideration in varied application areas like forecasting, fraud detection, modelling of photovoltaic arrays and power amplifiers. Previously, many algorithms such as the Back propagation algorithm, PSO and DE have been proposed for the training of CSPNN [10–14]. This work analyzes a FLANN like model coined as Chebyshev Polynomial Neural Network (CSPNN) to forecast the next adjacent day gold price using the past prices. CSPNN is a neural network, which consists of an Input Expansion Block (IEB), a weighted sum calculation block and an output calculation block (using the hyperbolic tangent function) in the output layer. In this work the random weights of the CSPNN forecaster are optimally updated by applying the evolutionary learning algorithms, that is, PSO and DE. Archival daily gold rates of UK/USD and MOS/USD dataset, from 16th August 2016 to 10th August 2019 are collected to validate the model.

The introductory section is followed by the following sections: Section 2 describes the architecture of CSPNN. Section 3 discusses the training techniques of CSPNN. Section 4 details the empirical setup and forecasting of gold prices. Section 5 draws a conclusion from the results.

2 Architecture of CSPNN

CSPNN is a one-hidden layer neural network like FLANN. Instead of having multi-hidden layers and multiple neurons, CSPNN is able to map the nonlinear relation between the in-values and out-values by widening the in-values using Chebyshev polynomials in the Input Expansion Block (IEB). Compared to FLANN, which uses trigonometric functions for input expansion, CSPNN depends on Chebyshev polynomials for the same and hence trains faster than FLANN. The actual benefit of the CSPNN is its clean architecture, rapid confluence and decreased computational overheads by expanding the dimensions of the in-values. Chebyshev FLANN provides improved efficiency over Multi-Layer Networks and basic FLANN.

The Chebyshev polynomials are expressed by $\mathrm{CS}_r(i)$, where r is the order and $-1 < i < 1$ is the parameter of the polynomial. These are orthogonal polynomials derived as the solutions to the Chebyshev equation. The recursive way of representing the polynomials are as follows:

$$\mathrm{CS}_{j+1}(i) = 2i\mathrm{CS}_j(i) - \mathrm{CS}_{j-1}(i). \tag{1}$$

The initial polynomials are represented as $\mathrm{CS}_0\ (i) = 1$ and $\mathrm{CS}_1\ (i) = i$. Using these initial values the higher expanded inputs are obtained recursively using Eq. (1).

Assuming the order r a q-dimensional input pattern $I = [i_1, i_2 \ldots i_q]^{\mathrm{T}}$ is extended to a x dimensional pattern CSI by the IEB as $\mathrm{CSI} = [1, \mathrm{CS}_1\ (i_1), \mathrm{CS}_2\ (i_1) \ldots \mathrm{CS}_r\ (i_1), \ldots, \mathrm{CS}_1\ (iq), \mathrm{CS}_2\ (iq) \ldots \mathrm{CS}r\ (iq)]^{\mathrm{T}}$, where $x = r*q + 1$. The expanded in-values along with the random weights are used to calculate the weighted input summation (WIS) as follows:

$$\mathrm{WIS} = \sum_{k=1}^{x} w_k \mathrm{CSI}_k. \tag{2}$$

The weighted input summation enters the output block which uses a nonlinear function (tan$h$ ()) to yield the predicted output. The predicted along with the actual output generates the error. This error is used to optimize weights by using any training algorithm.

3 CSPNN Training Techniques

Once the network architecture is ready for a required application, it must be trained.

The training begins with the initialization of random weights and further weights are tweaked to get the desired input and output relation.

The learning techniques of CSPNN include tuning of the weights optimally to decrease certain defined errors. So, training of CSPNN can be assumed as a way of optimization of an error measure. This section describes the working of two such evolutionary training algorithms for CSPNN.

3.1 PSO Learning

PSO is a population-based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behaviour of bird flocking or fish schooling. In this learning technique, each particle has a position and a velocity. The position represents a solution to the problem and hence needs to be optimized. The velocity refers to the speed with which a particle changes its position. The particle's position is revised using its own best position and the swarm's best position.

To use PSO for training of the CSPNN, the particle and velocity matrices are initialized with random weights. The fitness of each particle is calculated. The particles' own best (oBest) is initialized to the initial positions. The minimum fitness particle is chosen initially as the swarm's best (sbest). The positions and velocities are updated as follows:

$$V_{p,q}^{i+1} = \alpha V_{p,q}^{i} + a_1 r_{\mathrm{o}} \left(\mathrm{oBest}_{p,q}^{i} - P_{p,q}^{i} \right) + a_2 r_{\mathrm{s}} \left(\mathrm{sBest}_{p,q}^{i} - P_{p,q}^{i} \right). \tag{3}$$

$$P_{p,q}^{i+1} = P_{p,q}^{i} + V_{p,q}^{i+1}. \tag{4}$$

where α is the inertial weight, a_1 and a_2 are acceleration constants, r_o and r_s are random numbers within [0, 1] for own best part and swarm's best part, respectively. oBest is the particle's own best-known position and sBest is the swarm's best position.

The pseudocode for PSO is as follows:

for each particle
Initialize particle position and velocity with rand()
Repeat till allowed iterations or allowed error limit
for each particle
Calculate fitness(particle position)
if fitness is less than fitness(oBest)
Set oBest = current particle position
if fitness(oBest) is less than fitness(sBest)
Set sBest = oBest
for each particle
Calculate particle updated Velocity using equation (3)
Calculate particle updated Position using equation (4)
Select sBest as the frozen weight for testing the network.

3.2 DE Learning

DE is a learning algorithm introduced by Storn and Price in 1996. In DE, a population of candidate solutions are initialized randomly. The population performs three procedures, that is, mutation, crossover and selection on its candidates to get a new and fit generation. The fitness of each candidate is calculated and the candidates form the target vector. The candidates are mutated using the following mutation operation:

$$d_{c,k} = y_{c_1,k} + F(y_{c_2,k} - y_{c_3,k}) \quad \text{where} \quad c \neq c_1 \neq c_2 \neq c_3. \tag{5}$$

where $d_{c,k}$ represents the donor vector created for each candidate c after mutation, F is the mutation factor and c, c_1, c_2, c_3 are unique integers in the range [1, pop) and pop is the population size. After mutation the crossover operation is done to generate a trial vector $t_{c,k}$ as follows:

$$t_{c,k} = d_{c,k} \text{ if } r_1 \leq r_c \text{ or } k = k_{\text{rand}} \text{ else } t_{c,k} = y_{c,k}. \tag{6}$$

where r_c is the crossover rate, r_1 is any random number between 0 and 1.

After crossover, the trial vector (t) or the target vector (y) is chosen as a candidate for the next generation based upon the minimum fitness value. The pseudocode for DE is as follows:

Initialize the population.
Repeat till allowed iterations or allowed error limit
for each target candidate
Calculate fitness(target candidate)
for each candidate of population
donor = mutate(candidate) using equation (5)
for each candidate of population
trial = crossover(candidate,donor) using equation (6)
for each candidate of population
if(fitness(trial)is better than fitness(target))
Select trial as a candidate for next generation
Select the candidate with best fitness as the frozen weight for testing the network.

4 Testing of CSPNN with Output Analysis

To test the forecasting capability of the designed CSPNN, datasets of closing gold prices in dollars per troy ounce (oz) (1 troy ounce = 31.11 g) are gathered and pre-processed. The data is collected for London (UK/USD) and Moscow (MOS/USD) from 16th August 2016 to 10th August 2019. The number of specimens collected in both datasets is 765. The data is then normalized using the max-min technique to confine the data within 0 and 1 so as to limit the search space to the range 0 and 1. The normalized dataset is then arranged as in-values and out-values for training. To do so, the windowing technique is used. Here, a window of size 5 and a prediction horizon of size 1 is used. It signifies that the closing prices of five adjacent days are taken as in-values and their corresponding out-value is the next day (6th day) closing price. This technique reduces the dataset size to 760. Out of the 760 prices, the first 507(two-third) prices are used for training and the rest 253 prices are used for testing. The network is now designed to have five neurons representing the 5-day prices (5 dimensions) as input and one output neuron to depict the next day (6th day) gold price. This CSPNN then is trained using the two evolutionary algorithms, that is, PSO and DE. The weights are frozen after 200 iterations. The Root Mean Square Error (RMSE) is used as a fitness function here and is minimized in each iteration. The efficacy of the model is analyzed by using the following three error measures:

$$\mathrm{RMSE} = \sqrt{\frac{1}{D}\sum_{i=1}^{D}\left(\mathrm{agp}_i - \mathrm{pgp}_i\right)} \tag{7}$$

$$\mathrm{MSE} = \frac{1}{D}\sum_{i=1}^{D}\left(\mathrm{agp}_i - \mathrm{pgp}_i\right) \tag{8}$$

$$\mathrm{MAE} = \frac{1}{D}\sum_{i=1}^{D}\left|\mathrm{agp}_i - \mathrm{pgp}_i\right| \tag{9}$$

Table 1 Training and testing errors for both PSO and DE with varied expansion order for UK/USD dataset

Expansion order	Learning algorithm	Train RMSE	Train MSE	Train MAE	Test RMSE	Test MSE	Test MAE
2	PSO_avg	**0.0523**	**0.0028**	**0.0395**	**0.0351**	**0.0012**	**0.0272**
	DE_avg	0.0552	0.0031	0.0426	0.0377	0.0014	0.0298
	PSO_std	0.0052	0.0006	0.0043	0.0041	0.0003	0.0033
	DE_std	**0.0035**	**0.0004**	**0.0033**	**0.0025**	**0.0002**	**0.0019**
3	PSO_avg	0.0863	0.0082	0.0648	0.0627	0.0044	0.0490
	DE_avg	**0.0641**	**0.0042**	**0.0480**	**0.0465**	**0.0022**	**0.0364**
	PSO_std	0.0288	0.0059	0.0206	0.0225	0.0033	0.0170
	DE_std	**0.0079**	**0.0010**	**0.0054**	**0.0091**	**0.0009**	**0.0071**
4	PSO_avg	0.1177	0.0139	0.0928	0.1155	0.0137	0.0961
	DE_avg	**0.0730**	**0.0054**	**0.0562**	**0.0565**	**0.0033**	**0.0451**
	PSO_std	0.0096	0.0023	0.0091	0.0200	0.0046	0.0181
	DE_std	**0.0092**	**0.0014**	**0.0079**	**0.0113**	**0.0014**	**0.0096**

Minimum errors both average and standard deviation for each expansion order has been bold faced

where agp_i represents actual gold price and pgp_i represents predicted gold price and D is the number of data specimen collected. The efficiency of the CSPNN model greatly relies on the expansion order of the polynomials as it will reduce the time as well as help in mapping the nonlinear dependency of in and out-values for better prediction. Hence, the model has been simulated to execute 10 independent runs for both PSO and DE on 2, 3 and 4 order of expansion. The mean and standard deviation values are recorded for both the learning algorithms with varied expansion orders for both UK/USD and MOS/USD datasets. Table 1 depicts the minimum, average and deviation statistics for UK/USD and Table 2 depicts the same for MOS/USD dataset.

After analyzing Tables 1 and 2, it is clearly seen that for UK/USD dataset the PSO algorithm outperforms DE for second-order CSPNN, but DE performs well in third and fourth-order CSPNN. Also, for MOS/USD dataset DE performs better than PSO for all orders of expansions. As second-order CSPNN maps the nonlinearity between in and out-values less efficiently and 4th order CSPNN takes more time for computations, it is observed that a CSPNN of expansion order 3 using DE provides better prediction. Hence, using third-order CSPNN model, the two datasets are trained and tested. The results are depicted in Table 3; Figs. 1 and 2.

5 Conclusion

This work models CSPNN architecture and trains it with two evolutionary algorithms, that is, PSO and DE. It uses two past gold price data sets from London and Moscow

Table 2 Training and testing errors for both PSO and DE with varied expansion order for MOS/USD dataset

Expansion order	Learning algorithm	Train RMSE	Train MSE	Train MAE	Test RMSE	Test MSE	Test MAE
2	PSO_avg	0.0309	0.0010	0.0237	0.0514	0.0039	0.0336
	DE_avg	**0.0266**	**0.0007**	**0.0205**	**0.0468**	**0.0023**	**0.0313**
	PSO_std	0.0052	0.0003	0.0039	0.0373	0.0068	0.0166
	DE_std	**0.0036**	**0.0002**	**0.0030**	**0.0115**	**0.0012**	**0.0062**
3	PSO_avg	0.0533	0.0031	0.0414	**0.0778**	**0.0078**	**0.0524**
	DE_avg	**0.0311**	**0.0010**	**0.0241**	0.0779	0.0084	0.0463
	PSO_std	0.0166	0.0021	0.0131	**0.0439**	**0.0105**	**0.0195**
	DE_std	**0.0064**	**0.0004**	**0.0054**	0.0510	0.0118	0.0238
4	PSO_avg	0.0732	0.0057	0.0573	**0.1006**	**0.0103**	**0.0732**
	DE_avg	**0.0438**	**0.0020**	**0.0340**	0.1468	0.0469	0.0776
	PSO_std	0.0203	0.0035	0.0168	**0.0131**	**0.0029**	**0.0134**
	DE_std	**0.0079**	**0.0007**	**0.0073**	0.1678	0.1030	0.0642

Minimum errors both average and standard deviation for each expansion order has been bold faced

Table 3 Training and testing errors for both PSO and DE with expansion order 3 for both the dataset

Dataset	Learning algorithm	Train RMSE	Train MSE	Train MAE	Test RMSE	Test MSE	Test MAE
UK/USD	PSO	0.0554	0.0031	0.0418	0.0383	0.0015	0.0301
	DE	**0.0537**	**0.0029**	**0.0410**	**0.0357**	**0.0013**	**0.0290**
MOS/USD	PSO	0.0362	0.0013	0.0285	0.0386	0.0015	0.0306
	DE	**0.0250**	**0.0006**	**0.0192**	**0.0352**	**0.0012**	**0.0255**

Minimum errors for each dataset has been bold-faced

for training and testing. It analyzes the effects of different expansion order on the CSPNN gold forecasting using three error metrics. The simulation results clearly incite the use of a DE trained CSPNN with expansion order 3 as a better and efficient forecasting model for daily prediction of gold price.

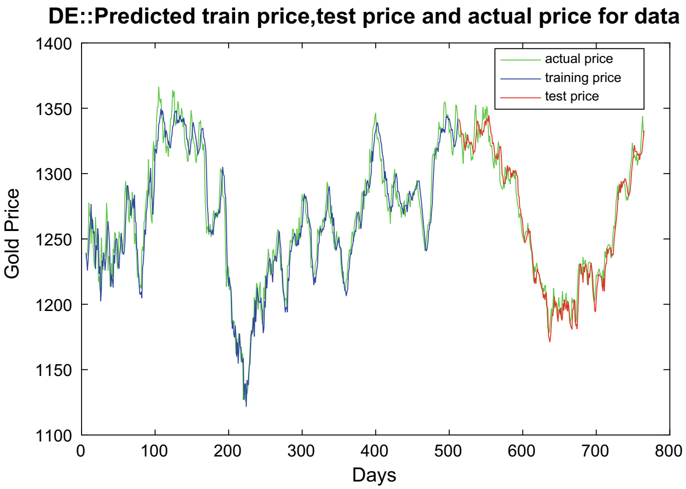


Fig. 1 Predicted train and test price with actual price using DE-CSPNN on UK/USD

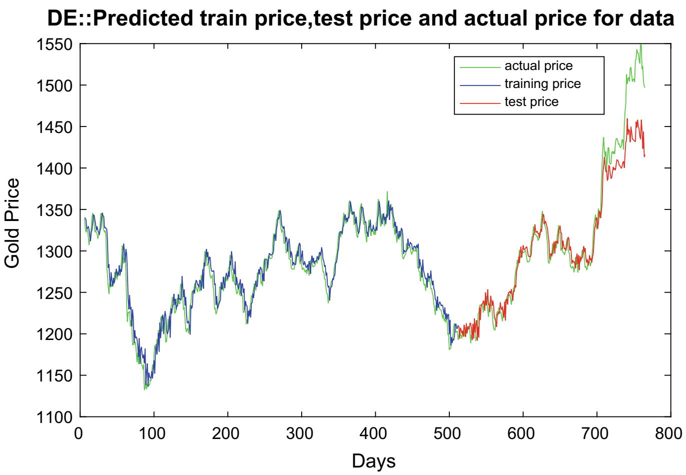


Fig. 2 Predicted train and test price with actual price using DE-CSPNN on MOS/USD

References

1. Dash, R., Dash, P.K., Bisoi, R.: A self adaptive differential harmony search based optimized extreme learning machine for financial time series prediction. Swarm Evol. Comput. **19**, 25–42 (2014)
2. Rout, A.K., Dash, P.K., Dash, R., Bisoi, R.: Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach. J. King Saud Univ.-Comput. Inf. Sci. **29**(4), 536–552 (2017)

3. Dash, R., Dash, P.K.: Prediction of financial time series data using hybrid evolutionary legendre neural network: Evolutionary LENN. Int. J. Appl. Evol. Comput. **7**(1), 16–32 (2016)
4. Kocak, H., UN, T.: Forecasting the gold returns with artificial neural network and time series. Int. Bus. Res. **7**(11), 139–152 (2014)
5. Mombeini, H., Yazdani-Chamzini, A.: Modeling gold price via artificial neural network. J. Econ., Bus. Manag. **3**(7), 699–703 (2015)
6. Kristjanpoller, W., Minutolo, M.C.: Gold price volatility: A forecasting approach using the artificial neural network–GARCH model. Expert Syst. Appl. **42**(20), 7245–7725 (2015)
7. Sivalingam, K.C., Mahendran, S., Natarajan, S.: Forecasting gold prices based on extreme learning machine. Int. J. Comput. Commun. Control **11**(3), 372–380 (2016)
8. Hadavandi, E., Ghanbari, A., Abbasian-Naghneh, S.: Developing a time series model based on particle swarm optimization for gold price forecasting. In: Third International Conference on Business Intelligence and Financial Engineering (BIFE), pp. 337–340 (2010)
9. Dash, R., Rautray, A., Rautray, R., Dash, R.: Gold price prediction using an evolutionary pi-sigma neural network. Int. J. Eng. Technol. **7**(4.5), 742–746 (2018)
10. Dash, R.: DECPNN: A hybrid stock predictor model using differential evolution and chebyshev polynomial neural network. Intell. Decis. Technol. Int. J. **12**(1), 93–104 (2018)
11. Patra, J. C., Thanh, N. C., Meher, P. K.: Computationally efficient FLANN-based intelligent stock price prediction system. In: IEEE International Conference in Neural Networks, (IJCNN), pp. 2431–2438 (2009)
12. Jiang, L. L., Maskell, L. L., & Patra, J. C. Chebyshev functional link neural network based modeling and experimental verification for photovoltaic arrays. IEEE International Conference in Neural Networks (IJCNN), (2012), 1–8
13. Li, M., Liu, J., Jiang, Y., Feng, W.: Complex-Chebyshev functional link neural network behavioral model for broadband wireless power amplifiers. IEEE Trans. Microw. Theory Tech. **60**(6), 1979–1989 (2012)
14. Mishra, M. K., Dash, R. A.: Comparative study of chebyshev functional link artificial neural network, multi-layer perceptron and decision tree for credit card fraud detection. In: IEEE International Conference in Information Technology (ICIT), pp. 228–233 (2014)

Weighted Particle Swarm Optimization with T-Distribution in Machine Learning Applications

Sonali Samal, Archana Sarangi, and Shubhendu Kumar Sarangi

Abstract Inspired by the biological development of evolution, Particle Swarm Optimization (PSO) is recognized as a simple and popular practice used for the optimization process. It is a population-based algorithm that has been applied to a broad number of problems in real-life and the foundation of this method is laid by Eberhart and Kennedy in 1995. This paper gives a layout of the advantage of putting a further modification supported by T-distribution in a modified pso for the furtherance of getting a more satisfactory outcome. Basically, the work surveys two modifications, i.e., moderation of traditional weight factor by an efficacious nonlinear inertia weight and again optimizing it with the T-distribution which conjointly delivers the global solution of higher quality and it is also finer in convergence. By analyzing the simulation results, it can also be verified that weighted pso with supported by T-distribution predominates all the cases of general pso.

Keywords Particle swarm optimization · Benchmark functions · T-distribution · Non linear inertial weight

1 Introduction

The fundamental concept of PSO has been growing rapidly as it deals with the optimization process which solves many real-life engineering problems by the method of maximization and minimization. In PSO, the algorithm started with randomly initializing the population or positions of each particle in the search space. Then it starts the searching for its global optima while it possesses many local optima in its

S. Samal · A. Sarangi
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India

S. K. Sarangi (✉)
Department of Electronics and Instrumentation Engineering, ITER, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: shubhendu1977@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_30

process of convergence towards its final global solution in case of multiplex problems. Hence it appraises as shortcomings for the entire PSO theory. To chuck out all the local optima get trapped in the process of general PSO, many researchers modified the approach of initializing some of the crucial parameters in these optimization algorithms. Increasing the diversity of the swarm and the convergence ability are the two impact factor on these swarm algorithms. By taking the consideration of the previous statistical analysis the aspect inertia weight has a vital responsibility in the convergence nature of PSO. Hence, among all the parameters inertia type weight plays a significant role in this algorithm as it controls the process of local search as well as global search. In this paper, the inertia weight has been modified with a better nonlinear weight factor and the T-distribution function is utilized. A good selection of parameters handles the searching space as well as gives a better set of personal best values and global best values which will upgrade the convergence power and eventually it gives a finer output than the previous one.

2 Optimization Techniques

2.1 Particle Swarm Optimization(PSO)

PSO [1] is a meta-heuristic optimization algorithm depending on the population. In PSO, particles are initialized, which are updated again and again to search for the optimum solution. The particle swarm optimization (PSO) algorithm is a comparatively innovative algorithm for universal optimization. It utilizes the knowledge of social communications demonstrated in a group of animals, e.g., fish swarming and bird flocking. Similar to the Genetic Algorithm, PSO is also regarded as an algorithm that depends on the population. The solutions provided by this algorithm are considered as particles but not individuals as in GA. The group of particles in a specified iteration is known as swarm. In the algorithm, the location of every particle is altered in accordance with fitness and place relating to the other particles in the group. The particles have a movement from beginning to end in the searching space with dynamism altering velocity. In mathematical domain [2], the structure of the algorithm for PSO can be specified as

$$\begin{aligned} \mathrm{VV}_i^{(k+1)} &= ww * \mathrm{VV}_i^k + \mathrm{CC}_1 * \mathrm{random}_1 * \left(\mathrm{particlebest}_i^k - SS_i^k\right) \\ &\quad + \mathrm{CC}_2 * \mathrm{random}_2 * \left(\mathrm{globalbest}^k - \mathrm{SS}_i^k\right) \end{aligned} \tag{1}$$

$$\mathrm{SS}_i^{k+1} = \mathrm{SS}_i^k + \mathrm{VV}_i^{k+1} \tag{2}$$

where, VV_i^{k+1} is the element's latest speed, *ww* is a parameter of inertia, VV_i^k is the element speed, SS_i^k is location of particle in *k*th iteration, CC_1 is accelerating constant for cognitive component, CC_2 is accelerating social constant, $\mathrm{particlebest}_i^k$ is earlier most excellent location of ith particle in kth iteration, globalbest^k is overall best location of entire population, random_1, random_2 are two uninformed numbers in range [0, 1] The component $CC_1 * \text{random}_1$ is an uninformed number with standardized distribution from 0 to CC_1 that quantify how deeply a particle believes its neighborhood most excellent velocity. Similarly $CC_2 * \text{random}_2$ is a random number with homogeneous distribution from 0 to CC_2 which signifies how much a particle faith in overall velocity. In the simple PSO, the population and the velocity are randomly taken for further procedure. Velocity is being updated in every step until the required generation gets achieved.

(1) Initialize the population and velocity on "d" dimension of a problem.
(2) For every particle calculate the fitness value.
(3) Make the comparison of the recent fitness value after computation with prior values of fitness and locate it as the personal utilizing effective value.
(4) Make an update for speed of particles using Eq. (1) along with location of particles utilizing Eq. (2).
(5) Revise the finest and global most excellent value of elements.
(6) Substitutions of updated particle vectors are done in the loop and repeat the process of updating of global best value.
(7) Iteration will keep on going till the permitted limit of iteration round.

2.2 Weighted Particle Swarm Optimization (WPSO)

In this case, the particle velocity is calculated by a modified expression that carries a modified weight parameter in place of a standard fixed value of weight which is constant throughout the execution of the algorithm. The modified weight parameter is given by [3, 4]

$$ww(t) = \frac{(\max\_w) - (\min\_w)}{1 + e^{-u*(t-n*\max\_\text{gen})}} + \min\_w \tag{3}$$

$$u = 10^{\log \max\_\text{gen}-2} \tag{4}$$

where $ww(t)$ is the new value of weight, max_w and min_w are the maximum value along with the minimum value of weight, respectively. In this case, max_gen indicates the maximum generation whereas n is the sigmoid constant and t is the current iteration. The execution sequence of the algorithm can be described as

(1) Initialization of the population and velocity for the problem.
(2) Start the generation loop.
(3) Before calculating the fitness values calculate the weight factor using the above equation.
(4) For every particle calculate the fitness value.

(5) Make a comparison for the recent fitness value after calculation with prior values of fitness in order to locate personal best with effective suitability value.
(6) Make the update for the speed of particles utilizing Eq. (1) along with the location of particles by Eq. (2) but utilizing the modified weight expression.
(7) Do the update for the particle finest and overall best with the modified weight expression.
(8) The evaluated output will act as the global best value.
(9) Replacement of the reorganized particle vectors as preliminary particle vectors will be done in the loop. Repeat the process of update of global best value as said in the prior steps.
10) Iteration will keep on going till the permitted limit of iteration round.

2.3 Weighted Particle Swarm Optimization with T-Distribution (MWPSO)

T-distribution which is also known as "student's T-distribution" was developed by William Sealy Gosset, which plays a wide role in many statistical analyses. It is a continuous probability distribution that arises when the population standard deviation is unknown and also where the sample size is small. It is a bell-shaped curve and symmetric in nature. It is almost similar to normal distributions but it has heavier tails. Tail heaviness will be defined from the no. of degrees of freedom taken in the problem. Higher the numbers of degrees of freedom more it will look like the normal standard deviation. In the normal distribution, i.e., Z-distribution the population standard deviation is taken into consideration but while implementing the T-distribution the sample or assumed standard deviation is taken into consideration

Degrees of freedom

In statistical analysis, the degrees of freedom are the total number of values that are free to vary in the data sample. It is used to measure statistical significance. The formula for degrees of freedom equals the size of total sample space—1.

Below is the Cumulative distribution function of T-distribution is [5]

$$p = F(x) = \int_{-\infty}^{x} \frac{\Gamma(\upsilon + 1/2)}{\Gamma(\upsilon/2)} \frac{1}{\sqrt{\upsilon\pi}} \frac{1}{\left(1 + \frac{t^2}{\upsilon}\right)^{\frac{\upsilon+1}{2}}} \mathrm{d}t$$

where ν is recognized as degrees of freedom and $\Gamma(.)$ is the gamma function in the interval of $[-\infty, x]$

Steps of execution for MWPSO

(i) Initialization of population and velocity randomly with the given assigned population and dimension size.

(ii) Initialization of all the parameters related to modified particle swarm optimization and a special weight factor will be introduced in the velocity to obtain a better result.
(iii) The fitness value is calculated by evaluating the given benchmark function.
(iv) Initialization of overall most excellent location value.
(v) Till the maximum iteration, the subsequent steps will get repeated.
(vi) Now apply T-distribution to the algorithm and calculate the value which will be assigned as the global best value as of now.
(vii) Update the global best and store it in a matrix.
(viii) Update the position and velocity.
(ix) Again using the objective function, compute the overall finest value.
(x) Iteration continues till it reaches the limiting value.

3 Result Analysis

The benchmarking function selected (Table 1) for utilization in the simulation experiment is suitable for real-world engineering applications. The intention of the simulating experiment is the minimization of the function values. For the verification of the novel algorithms in addition to the existed base modified version of PSO, the usual parameters are in use as described. For WPSO and MWPSO, $CC_1 = CC_2 = 2$, $ww = 0.72{,}984$, Particle number = 120. The dimension of the used function [6] in the simulation experiment is taken as 08. All experimenting procedure has been repeated for 40 trials. The mean value of the best particles during the optimization experiments has been recorded. The record was also taken for the finest and most awful values created in all the function evaluations in the complete research. There is no requirement for considering the PSO algorithm in the simulation process as performance of the modified referred algorithm, i.e., WPSO is better than PSO in referred literature.

The results of simulations of every benchmark function for WPSO and MWPSO algorithms are specified in Table 2 consecutively. It can be demonstrated that modification of a weighted particle swarm optimization algorithm with a T-distribution

Table 1 Benchmark function

Function name	Function expression
Happycat function	$f_1(x') = \left\lvert \sum_{i=1}^{DD} x_i'^2 - DD \right\rvert^{1/4} + \left(0.5 \sum_{i=1}^{DD} x_i'^2 + \sum_{i=1}^{DD} x_i'^2\right)/DD + 0.50$

Table 2 Performance analysis for function

Algorithm	Best	Worst	Mean
WPSO	1.7526	1.9964	1.9558
MWPSO	0.7759	0.7834	0.7764

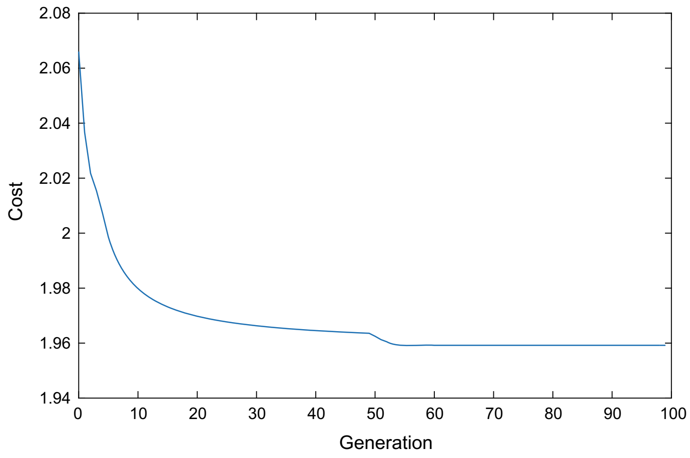


Fig. 1 Cost versus generation profile for function utilizing WPSO

function with 40 self-sufficient runs performing better while a comparison is done to WPSO in terms of best, mean, worst values. The finest values produced in WPSO for the standard functions utilized in simulation is 1.7526, but when the simulation is done for the newly proposed MWPSO utilizing the identical function, the finest value is originated as 0.7759. The best value when compared proved the better utility of projected MWPSO. Similarly, improvement of the results by the new modified MWPSO algorithm for the benchmark function can be easily visualized by the comparison of mean, worst values described in Table 2. The mean values obtained during the entire simulating experiment for the new modified algorithm (MWPSO) are found as 0.7764 which shows the precision of the measurement by this algorithm. But the other algorithm, i.e., WPSO provides mean values of 1.9558 for the same function. The worst values obtained during the simulating experiment also agree with the fact that the proposal of a new modified version of the algorithm is productive. Figure 1 in addition to Fig. 2 represents convergence profile of the new modified mutated weighted particle swarm optimization along with weighted particle swarm optimization. In all the figures it can be visibly verified that the outline of convergence of the new proposed modified algorithm is much better when a comparison is done with previously modified PSO, i.e., WPSO. It can also be visibly established that in almost all figures the new modified algorithm MWPSO converges at a quicker rate in comparison to previously existed modified algorithms. Figures 3 and 4 provide a better analysis of algorithm performance with a variation of population size. The bar plot in Fig. 3 is for WPSO which shows the lowest mean value when the population size is close to 125. But Fig. 4 provides an almost flat response for the mean values of with a very small reduction at 125. This shows that the proposed algorithm performance is not very dependent on the population size. This experiment also verifies that the novel modification has the effectiveness over the older one for any size of population.

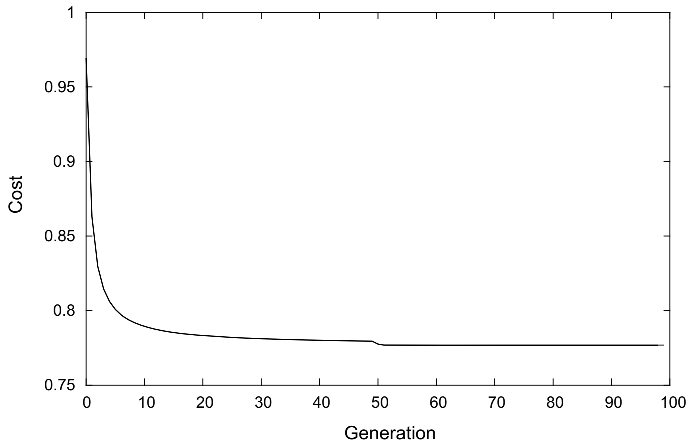


Fig. 2 Cost versus generation profile for function utilizing MWPSO

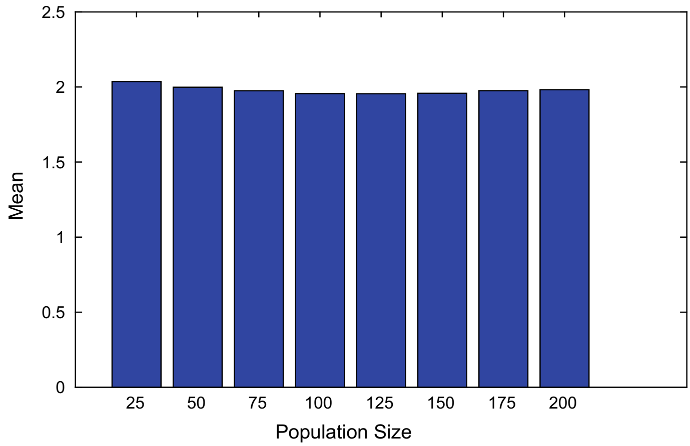


Fig. 3 Mean versus population for function for WPSO

4 Conclusion

In this paper, an upgraded version of the weighted PSO variant is projected to enhance the performance of the existing modified version of standard PSO. The proposed approach is known as mutated weighted particle swarm optimization (MWPSO). This algorithm combines the finest features present in the previously existing modified PSO algorithm with the T-distribution function. This proposal of improving the exploitation by the T-distribution concept is found fruitful after the experiment is done with the proposals by utilizing the standardized benchmarking functions. This

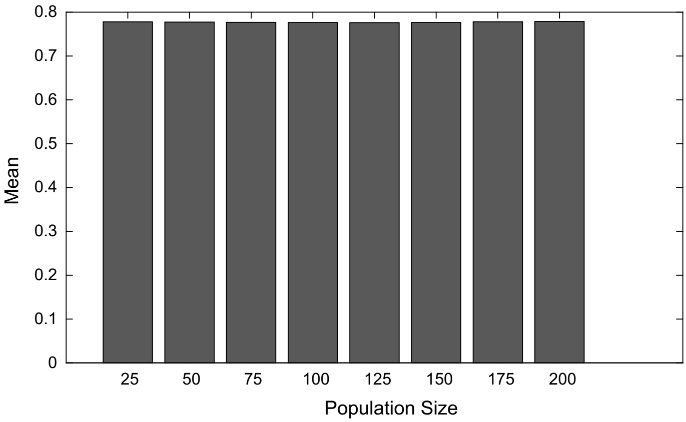


Fig. 4 Mean versus population for function for MWPSO

proposal is also very simple thereby not provides much complexity in the computation. The new modified and improved algorithm provides a sense of balance between investigation and exploitation. The algorithm proves its usefulness in terms of all the statistical parameter standards when a comparison is made with the previously weighted PSO algorithm on testing by utilizing typical benchmarking functions. Since the newly modified version provides a better result at a lower value of population, there is no need of taking higher population size. In case of the same population size, the small amount of additional time required by the innovative customized algorithm can be effortlessly remunerated by superior exactness in the results. As a result, the innovative approach of modification in weighted PSO provides better-quality outcomes and can be made applicable for a lot of applications in machine learning for training of neural networks, etc.

References

1. Kennedy, J., Eberhart, R.C.: Particle swarms optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
2. Ho, S.L., Yang, S., Ni, G., Lo, E.W.C., Wong, H.C.: A particle swarm optimization-based method for multi objective design optimizations. IEEE Trans. Magn. **5**, 1756–1759 (2005)
3. Tian, D., Shi, Z.: MPSO: Modified particle swarm optimization and its applications. Swarm Evol. Comput. **41**, 49–68 (2018)
4. Chen, G., Huang, X., Jia, J.: Natural exponential inertia weight strategy in particle swarm optimization. In: Proceedings of the World Congress on Intelligent Control and Automation (WCICA'06), pp. 3672–3675 (2006)
5. Walpole, R.E., Myers, R.H., Mayers, S.L.: Ye, K.E.: Probability and Statistics for Engineers and Scientists, 9th edn. Person publications, India

6. Chen, Q., Liu, B., Zhang, Q., Liang, J.J., Suganthan P.N., Qu, B.Y.: Problem definitions and evaluation criteria for CEC 2015 special session on computationally expensive single objective optimization, Technical report (2014)

A Novel Approach for Breast Cancer Data Classification Using Deep Forest Network

Bishnupriya Panda, Shrabanee Swagatika, Sipra Sahoo, and Debabrata Singh

Abstract This work presents a deep forest network-based breast cancer classification system. Post application of principal component analysis (PCA) on the dataset reduced the dimensionality of data from nine dimensions to four dimensions to keep most uncorrelated features. Subsequent application of tenfold cross-validated deep forest-based classifier resulted in improved prediction accuracy. The proposed model is compared with contemporary neural network systems. The proposed model outperforms the contemporary neural network model with substantial prediction accuracy of 99.11%. The major contribution of this work is development and application of deep forest model for breast cancer classification.

Keywords Principal component analysis · Deep network · Deep forest network · Breast cancer

1 Introduction

Breast cancer has been a major concern for women health and hygiene worldwide. After lung cancer, it is the major cause of women's death. Worldwide it is 12% among all cancers and 25% for cancers among women. The most widely applied method for breast cancer detection is mammography [1]. Most challenging issue

B. Panda (✉) · S. Swagatika · S. Sahoo
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Odisha, Bhubaneswar 751030, India
e-mail: bishnupriyapanda@soa.ac.in

S. Swagatika
e-mail: shrabeneeswagatika@soa.ac.in

S. Sahoo
e-mail: siprasahoo@soa.ac.in

D. Singh
Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Odisha, Bhubaneswar 751030, India
e-mail: debabratasingh@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_31

on mammography-based detection is correct interpretation. Most radiologists differ in their interpretation of mammography [2]. Another widely used method of breast cancer detection is fine-needle aspiration cytology (FNAC) with success rate of 90% only. In time, accurate interpretation and detection of breast cancer can save millions of lives worldwide. Hence, there remains an urgent need of automated, reliable and unbiased machine learning-based system which can assist radiologists in early decision making.

Machine learning and data mining have played a crucial rule in many real-life problems spanning from game playing, stock price prediction, protein classification, microarray data analysis to robot-assisted surgery and many more. Successful application of artificial neural networks (ANN) to a range of classification problems along with breast cancer [1] has been successfully demonstrated previously. A hybrid architecture of neural network and decision trees-based model was suggested by [3]. A combined fuzzy-artificial immune system and k-NN algorithm was proposed by [4] for breast cancer diagnosis. In [5], authors have compared the performance measurements of a spectrum of classifiers such as multilayer perceptron (MLP), support vector machine (SVM), probabilistic neural network and classical recurrent networks. Authors in [6] have also compared a set of classifiers such SVM with radial basis function [7] kernel, radial basis function neural networks [8] and random forest [7].

2 Materials and Methods

2.1 Dataset

We have used the publicly available Wisconsin breast cancer datasets [9] in this research which can be summarized as two classes out of which 65.5% are malignant and 34.5% are benign. The details of the dataset are summarized in Table 1.

2.2 Principal Component Analysis and Dimensionality Reduction of Wisconsin Breast Cancer Dataset

High-dimensional data are very common in biological and many other areas of scientific research. PCA is an unsupervised learning approach, similar to clustering, and has no prior knowledge about data distribution. In summary, PCA [10] can be defined as a transformation procedures which reduces a group of highly redundant features to a set of uncorrelated features also called as principal components (PCs). Principal components are always sorted in terms of their captured variability. The component that captures the highest variability becomes the first principal component and so on. The first few PCs mostly catch the most variability present in the data showing the

Table 1 Brief description of used dataset

Attribute ID	Brief description	Range of value	Mean	Standard deviation
1.	Thickness of clump	1–10	4.41	2.81
2.	Cell size	1–10	3.128	3.04
3.	Cell shape	1–10	3.198	2.96
4.	Marginal adhesion	1–10	2.79	2.86
5.	Single epithelial cell dimension	1–10	3.2	2.2
6.	Bare nuclei	1–10	3.459	3.65
7.	Bland chromatin	1–10	3.428	2.43
8.	Cell nucleoli normal	1–10	2.89	3.04
9.	Mitosis	1–10	1.585	1.7

major orientations of data dimensions. All other features present in the original data can be regenerated from these PCs which are orthogonal to each other. Wisconsin breast cancer dataset is a low-dimensional (9) dataset. Still, further reduction of this dataset helps in removing correlated data from the dataset which is always been considered as a good practice in machine learning community.

2.3 Neural Networks

Machine learning community is consistently striving to develop broad categories of network models for prediction and classification problems, spanning from simple architectures (FLANN) [11] to extremely specialized networks (deep networks) that handle specific problems. Every neural network model varies in its topology in terms of the number of hidden layers and nodes in each layer. Moreover, every architecture is special in terms of the learning algorithm employed to train the network. The most prevailing model is the multilayer perceptron (MLP), which has a feed-forward arrangement of layers of nodes. Every MLP architecture subsists of one input layer, one output layer and one or more hidden layers. The layers sandwiched between the input and output layers are known as hidden layers mostly responsible for real computation involved in the architecture. A simple architecture of a MLP network is given in Fig. 1.

Nodes between the layers are fully connected initially with random weights. In addition to the connecting weights, both the hidden layer and nodes of output layer receive additional bias signals. Bias is also weight with values 1. Nodes in the input and output layer are determined by the data dimensionality. If the dataset has N features, then the node count in the data absorption layer is usually N. If the problem is a classification problem, then the node count in the model output layer is decided by the number of clusters or classes present in the data otherwise only one node in the output layer if it is a regression or prediction task. The role of the hidden layer is most

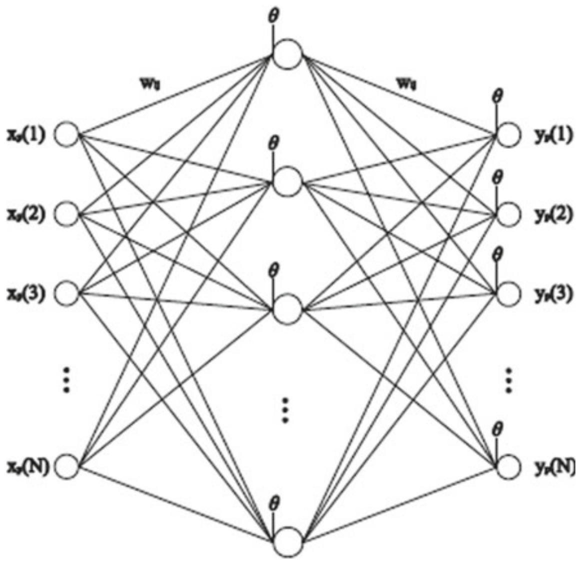


Fig. 1 A simple multilayer perceptron network with input, hidden and output layer

crucial in MLP as it maps the data in input space from a low-dimensional perspective to a relatively larger space and converts a nonlinear task in lower coordinate system to a linear task in a higher coordinate system. However, predefining the counts of hidden layers and nodes in the hidden layers is still an area of active research as more layers and uncontrolled number nodes may lead to the problem of data overfitting. Overfitting of data makes model to lose its generalization ability. Therefore, many different optimization algorithms like NSGA-II [12] are used for network structure optimization, thereby preventing the network from overfitting. Apart from this, most widely used algorithm to train a multilayer network is backpropagation algorithm. As per backpropagation algorithm, error incurred at the final layer of the network is not solely due to the nodes in the output layers. Hence, there should be an even distribution of error management across all the nodes of different layers except the input layer nodes. In addition to the backpropagation algorithm, different optimization algorithms are used for faster convergence of the network.

2.4 Architecture of Deep Forest Network

Deep learning [13] has been widely applied in various domains in the last decade. A huge success of deep learning in speech processing and image classification made them much popular in machine learning community. Most deep networks are built on multiple layers of neural networks as their building blocks mostly trained by backpropagation algorithm and its variants. More advanced optimization techniques like Adam [14] are also used for faster training and convergence. The primary

Table 2 Comparative analysis of deep networks and deep forest in terms of hyperparameters optimization

Deep networks and variants	Deep forest
Activation functions: tanh, sigmoid, softmax how to decide Architectural issues: 1. How to decide the layers' statistics? 2. How to decide kernel size? 3. How to decide feature mapping? Network performance optimization: 1. How to decide learning rate and momentum? 2. How to decide dropout percentage for each layer? 3. How to initialize weights? • Glorot uniform • Linear • Zeros	Type of forest: completely random forests No. of trees in forest: maximum 500 No. of forests in multi-grained scanning-2 Tree growth: till leaf nodes No. of forests in cascade-8 Tree growth: till leaf nodes No other hyperparameters optimization required

inherent issue in deep neural networks is to optimize a set of hyperparameters for optimal performance of the network. Secondly, deep networks require a huge amount resource in terms of training time and data size. The performance of deep networks is hugely affected for small- and even medium-level datasets. Thirdly, deciding parameters like number of layers, dropout percentage, learning rate and regularization criteria are most tricky tasks while implementing a deep network. Moreover, in many research areas, there may not be huge amount of data to train a deep network. In such scenario, deep networks have severe limitations. Recently, many deep networks like long short-term memory networks have gained popularity with enhanced complexity. Complexity of deep network-based models and their variants is not data-dependent but architecture-dependent, which can also be cited as limitation of deep networks. A relative comparison between deep networks and deep forest [15] in terms of hyperparameters optimization is shown in Table 2.

3 Experiment

Deep network relies on representation learning where each layer process the raw data received from its previous layer. Deep forest also employs a similar cascade of forests where every layer of cascade receives input from its previous layers as depicted in Fig. 2. Each layer in deep forest is an ensemble of ensembles, each ensemble representing a set of decision tree forests which also ensures diversity in the architecture of the deep forest. Construction of random forests is achieved by random selection of features, and trees in the forests grow till the leaf nodes.

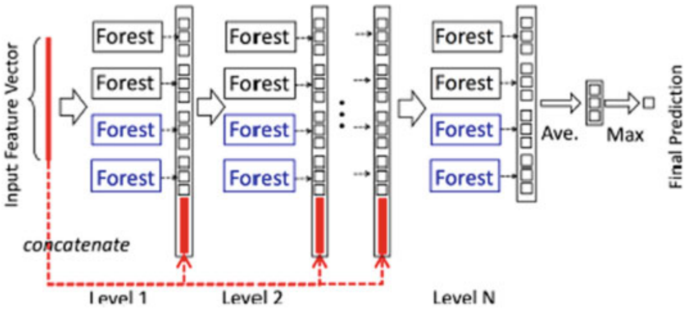


Fig. 2 A cascaded deep forest structure

Table 3 Architectural details of cascaded forest model

Model	Model type	No. of forests	No. of trees in each forest	Random feature selection criteria
Deep forest	Cascaded forest architecture	2	500	$\sqrt{(d)}$, d is size of input

Each layer of the cascade has two forests in black and two forests in blue. All the forests are completely random.

Random selection features can be obtained in a spectrum of ways. Here, we have considered no of features $= \sqrt{(d)}$, where d is the shape of the input feature space. Number of features can also be selected as no of features $= \log 2(d)$. The architectural details are given in Table 3.

4 Results and Discussions

The performance of deep forest is shown in Table 4 and has been compared with different variants of neural networks with different combinations of the input data. The results were validated with tenfold cross-validation. For our purpose, we have

Table 4 Performance comparison of reported model with neural network (NN)-based model with different input data sizes

Classifier	No. of epochs	Correctly classified	Misclassified	Percentage accuracy
NN + All features	61	216	11	95.2
NN + 8 features	44	221	6	97.4
NN + 4 features	33	217	10	95.6
Deep forest	**NA**	**225**	**2**	**99.11**

The bold values represent it shows superiority as compared to other approaches.

considered only four-dimensional PCA reduced representation of data. The results of neural network-based model are shown in Table 4.

5 Conclusion and Future Work

In this study, an automatic breast cancer diagnosis system is developed using deep forest. Despite of low dimensionality of input features, we have further reduced it to four dimensions in order to keep only most uncorrelated data. The performance of any classifier is significantly affected if features are not selected in a suitable manner. Unwise selection of features sometimes resulting in overtraining and the model loses overall generalization ability. Principal component analysis is a decade-old feature extraction and reduction technique. The proposed deep forest model eliminates the limitations of deep networks which requires a number of hyperparameters for optimal result. Even in the traditional neural network architecture parameters like hidden layers, nodes in the hidden layers, convergence rate and momentum are to be decided. These issues prevent both deep networks and traditional neural networks to be fully automatic. On the other hand, very few parameters in deep forest make them a natural choice to be deployed as a fully automatic model.

References

1. Chou, S.M., Lee, T.S., Shao, Y.E., Chen, I.F.: Mining the breast cancer pattern using artificial neural networks and multivariate adaptive regression splines. Expert Syst. Appl. **27**(1), 133–142 (2004)
2. Elmore, J.G., Wells, C.K., Lee, C.H., Howard, D.H., Feinstein, A.R.: Variability in radiologists' interpretations of mammograms. N. Engl. J. Med. **331**(22), 1493–1499 (1994)
3. Jerez-Aragonés, J.M., Gómez-Ruiz, J.A., Ramos-Jiménez, G., Muñoz-Pérez, J., Alba-Conejo, E.: A combined neural network and decision trees model for prognosis of breast cancer relapse. Artif. Intell. Med. **27**(1), 45–63 (2003)
4. Şahan, S., Polat, K., Kodaz, H., Güneş, S.: A new hybrid method based on fuzzy-artificial immune system and k-nn algorithm for breast cancer diagnosis. Comput. Biol. Med. **37**(3), 415–423 (2007)
5. Ubeyli, E.D.: Implementing automated diagnostic systems for breast cancer detection. Expert Syst. Appl. **33**(4), 1054–1062 (2007)
6. Chaurasia, V., Pal, S.: Data mining techniques: to predict and resolve breast cancer survivability (2017)
7. Panda, B., Majhi, B., Thakur, A.: An integrated-OFFT model for the prediction of protein secondary structure. Curr. Comput. Aided Drug Des. (2018)
8. Panda, B., Majhi, B.: Development of novel multi-objective based model for protein structural class prediction. Handbook of Research on Computational Intelligence Applications in Bioinformatics, pp. 76–107. IGI Global (2016)
9. Wolberg, W.H., Mangasarian, O.: Breast cancer Wisconsin (original) data set. University of California, UCI Machine Learning Repository (1989)
10. Mackiewicz, A., Ratajczak, W.: Principal components analysis (PCA). Comput. Geosci. **19**, 303–342 (1993)

11. Panda, B., Mishra, A.P., Majhi, B., Rout, M.: Prediction of protein structural class by functional link artificial neural network using hybrid feature extraction method. In: International Conference on Swarm, Evolutionary, and Memetic Computing, pp. 298–307. Springer, Cham (2013)
12. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.A.M.T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
13. Panda, B., Majhi, B.: A novel improved prediction of protein structural class using deep recurrent neural network. Evol. Intell. 1–8 (2018)
14. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014). arXiv preprint arXiv:1412.6980
15. Zhou, Z.H., Feng, J.: Deep forest: towards an alternative to deep neural networks (2017). arXiv preprint arXiv:1702.08835

A Survey on Deep Learning: Convolution Neural Network (CNN)

Madhusmita Sahu and Rasmita Dash

Abstract Deep learning is a subfield of machine learning and plays a vital role in the area of image processing, natural language processing, computer vision, etc. As compared to traditional machine learning methods, it has a strong ability of self-learning and self-debugging. Convolution neural network (CNN) is the most widely used technique of deep learning for better feature extraction from large datasets. Many researchers adopted CNN for object classification, face recognition, automatic handwritten, etc. In this paper, the detailed concepts behind CNN are discussed with their broad applications.

Keywords Artificial neural network (ANN) · Machine learning · Deep learning · Convolution neural network (CNN)

1 Introduction

Deep learning [1] is the type of data mining technique, which is the branch of machine learning, and again, machine learning is the subset of artificial neural networks (ANN). ANN is the type of neural network which tries to mimic the human brain. Machine learning is automatically learning itself and predicts the output depends on input data. Whatever the error incurred that is tried to be minimized by the programmer. The objective of the programmer to find out the best accuracy by providing what types of input data to train the model with minimum or zero error. But deep learning algorithms train the model with an enormous amount of datasets, and the model has rectified any errors by itself without the help of explicit programmers.

M. Sahu (✉)
Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: madhusmitasahu@soa.ac.in

R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: rasmitadash@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_32

Because of increasing processing power exponentially, nowadays, the deep learning plays a vital role in many areas. Deep learning networks are widely used to predict with the highest amount accuracy as compared with other machine learning tools. Different types of architecture [1] are listed below.

Deep Neural Network (DNN)—This is the type of neural network with multiple hidden layers in between input and output layers, and they can be used to model and process any nonlinear relationships.

Deep Belief Network (DBN)—This is a type of multilayer belief network which uses a contrastive divergence algorithm for the first layer to collect the features followed by some activation functions in the next layer, and finally, DBN is trained with the achievement of first hidden layer.

Recurrent Neural Network (RNN)—Because of their internal memory, RNN's can remember important things about the input they received, which allows them to be very precise in predicting what's coming next. This is why they're the preferred algorithm for sequential data like time series, speech, text, financial data, audio, video, weather and much more. Recurrent neural networks can form a much deeper understanding of a sequence and its context compared to other algorithms.

2 How Deep Learning Algorithm Differs from Artificial Intelligence (AI) and Machine Learning

Many peoples have misconceptions that deep learning and machine learning are concept of the same category and related to AI. But AI is one of the broad areas in which machine learning is part of it and again deep learning is also a subpart of machine learning. They are differentiated from each other by some specific qualities, but the main objective is to achieve AI that is enabling any program to behave like a human being. Deep learning has the capability of rectifying errors itself, but machine learning programmers have to involve in debugging explicitly if any problem exists. Machine learning is widely used in many application programs and is being used to give suggestions to their customers for choosing any TV serials, movies, etc., by considering their past preferences. Deep learning is acquiring more and more data to train its model, and it is independent of others for debugging. The automatic car driving system is a good example of deep learning. Both machine learning and deep learning are used to achieve AI.

2.1 Difference Between Machine Learning and Deep Learning

Machine learning requires small dataset to train its model, while deep learning requires vast number of datasets to train its model. Implementation of machine learning can be done in low-end machine, and it requires less time for training, but it consumes more time for testing. But, in case of deep learning, implementation can be done in high-end machine and consumes less time for testing, but requires longer time for training.

3 Deep Learning CNN Model

Convolution neural networks (CNNs) [2] are the most popular deep learning architecture. Mainly, CNN is used in image-related problems [11]. CNN can be used in Language Modeling, Sentiment Analysis, Language Translation, and more. The use of CNN is widely increased because of its capability of extracting important patterns from the image without the involvement of human beings. For example, for a given image set of dog and cat, CNN trained a model having a large number of datasets, and the model automatically identifies that the particular image is a dog or cat by having some important features. CNN model is a very efficient and pure model, and it has many layers of convolution, pooling and fully connected layers. CNN [3, 4] shares the weight parameter in fully connected (FC) layers and so reduces the time overhead in the calculation of every weight.

3.1 History Behind CNN

Two neurophysiologists—David Hubel and Torsten Wiesel—in 1959. In their publication, entitled "Receptive fields of single neurons in the cat's striate cortex", described core response properties of visual cortical neurons as well how a cat's visual experience shapes its cortical architecture. The duo ran some pretty elaborate experiments. They placed electrodes into the primary visual cortex area of an anesthetized cat's brain and observed, or at least tried to, the neuronal activity in that region while showing the animal various images (Fig. 1). A few months into the research, they noticed, rather accidentally, that one neuron fired as they were slipping a new slide into the projector. This was one lucky accident! After some initial confusion, Hubel and Wiesel realized that what got the neuron excited was the movement of the line created by the shadow of the sharp edge of the glass slide.

The researchers discovered some neurons present in the primary visual cortex as a combination of simple and complex neurons, and they are activated by visualizing some oriented edges [1]. This feature is used as the primary concept in deep learning. For this reason, the first digital image scanner was invented. In 1959, an apparatus was developed by Russell Kirsch along with his colleagues for transforming images into grids of numbers so that the machine can understand the images.

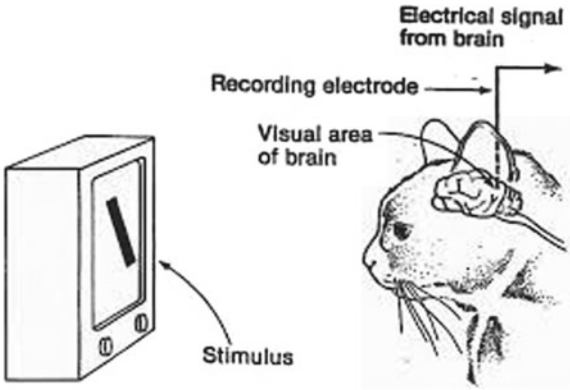


Fig. 1 Experiment on cat's brain

Table 1 ANN versus CNN

ANN	CNN
ANNs take a vector of inputs and produce an output through another hidden layer vector fully connected to that input	But CNN exploits the structure of image through different layers of it which is discussed in the next section
ANN consumes more time in estimating weights. Ex: $224 \times 224 \times 3$ weights need to be calculated for a single neuron in the output layer for a RGB image of size 224×224	But in CNN, less time incurred, as it is leading sparse connections between input and output neurons
There is no parameter sharing between output neurons which take place	There is a parameter sharing between output neurons of CNN

3.2 Reason Behind Choosing of CNN Over ANN

Nowadays, CNN is chosen for every application. The CNN has some architectural advantages over ANN, which are discussed in given Table 1.

3.3 CNN Architecture

CNN takes an image in Fig. 2 in three dimensions (width, height, depth). So, neurons are arranged in that manner. Each layer of CNN takes 3D input and transforms it into the 3D output of neuron activations. If it is an RGB image, then depth is 3, and height and width will be the dimensions of the image. CNN works well on image. CNN consists of convolution, pooling layers and fully connected networks. Each layer of CNN is discussed in the next subsection.

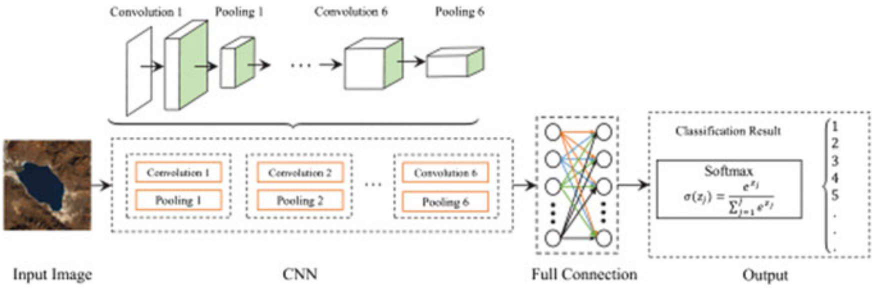


Fig. 2 Image shows the architectural overview of CNN

3.3.1 Convolution

Convolution layer is the vital building block of CNN. It acts as a convolution of two inputs to produce an output. This is the first layer that is applied to input data to get a feature map. It takes input data and a kernel. If the input is an image, then the kernel will be a 2D array consisting of 0 and 1 s. The corresponding output will come out according to the shape of the kernel. The kernel/filter moves to the right of the input image with a certain Stride Value until it parses the complete width. Moving on, it hops down to the beginning (left) of the image with the same Stride Value and repeats the process until the entire image is traversed. Then this would be the input for the next stage.

The convolution $\mathrm{conv}(I \cdot k)$ is a dot product of the input image, *I,* and the convolution kernel, *k*. The output is a convolved feature map, F_c. Formally, F_c is obtained by

$$\begin{aligned} F_c &= \mathrm{conv}(i, j) \\ &= (I \otimes K)(i, j) \\ &= \sum m \sum n \; I(m, n)k(i - m, j - n) \end{aligned} \tag{1}$$

$\otimes$ denotes a 2D discrete convolution operator in Eq. 1, which shows that the convolution kernel "*k*" spatially slides over the input image *I* to compute the element-wise multiplication and sum to produce an output a convolved feature map F_c. For example, let us consider the convolution operation of input image 5×5 with the 3×3 kernel to produce a 3×3 convoluted feature map. For that, the multiplication of each element of input data with each element of the kernel is done and the final output of each multiplication operation is stored in the feature map of 3×3. The size of the feature map depends on the size of the kernel you have taken. If the image is an RGB-colored image, then the multiplication operation is carried out 3 times as a multiplication of red image with the kernel, multiplication of blue image with kernel and multiplication of green image with the kernel. Finally, all three outputs are combined to form convolution output.

3.3.2 NonLinearity

Any neural network becomes powerful when it has some nonlinearity in the input. Each output of the convolution layer is gone through the activation function. ReLU activation units are used in the output of the convolution layer and stores the output ReLU operation in the feature mapped array (F_T) in Eq. 2.

$$F_T = \text{ReLU}(x_i) = \max(0, x_i) \tag{2}$$

3.3.3 Stride and Padding

How the convolution filter moves on the input image is decided by stride which specifies how many numbers of locations the filter slides on the image. The size of the convolution filter must be less than the input image size. If it is not matched, then padding of zero values is added in the image to make a match with the size of the convolution filter. Generally, the padding on CNN is useful as it is used to preserve the size of the feature map. Every layer of CNN the input is shrunk, so padding is used to get back the desired dimension. The stride of 1 moves the filter one pixel at a time, and similarly, the stride of 2 moves the filter two pixels at a time. Each convolution layer takes input volume in Eq. 3 to produce output volume whose size is decided by three hyperparameters such as depth (no. of filters used), stride and zero padding. How many neurons (say N) will be fitted in the output layer of the convolution layer can be computed as the formula.

$$N = (I_W - C_F + 2Z_P)/(S_z + 1) \tag{3}$$

where I_W = Size of input volume, C_F = Size of the filter, Z_p = Amount of zero padding, S_z = Applied stride.

3.3.4 Pooling

Pooling is the next layer of the convolution layer, which is used to reduce the dimension. It is useful for reducing the number of parameters that yield less training time. Padding is used to combat overfitting. It is used to downsample each feature map with height and width, but depth remains the same. There many types of pooling available; few of them are max pooling and average pooling which are the most commonly used. Max pooling yields the maximum value among all values in the pooling window. The average pooling (F_p) can be computed as in Eq. 4.

$$\text{Average pooling}\left(F_p\right) = \text{pool}(i, j) = \frac{1}{M}\sum mXi, j \tag{4}$$

3.3.5 Fully Connected (FC)

The fully connected layer is the last layer of CNN architecture. This is the same as ANN architecture. This Fc layer takes input from other layers and transforms them into specific no. of classes which are already decided by the network. The output layer of the FC layer is computed for error calculation. Then, a loss function (SVM/Softmax) is defined to compute the gradient of error. These errors are propagated backwardly to update weights and bias in backpropagation neural network. In one forward and backward pass, one cycle is completed for training.

4 CNN Application

Because of CNN popularity, it has many applications and used in extensive manner. Some of them are discussed below.

CNN can be trained by taking some medical images to identify particular disease. Kido et al. in 2018 [2] used an image-based CADX with help of CNN, which was used to differentiate different kinds of lung abnormalities such as nodules and diffuse lung diseases with a mean accuracy of 95.2% without data augmentation and 99.4% with data augmentation. Lu et al. in 2019 proposed an effective framework for script identification by integrating local CNN and global CNN, both of which are based on ResNet-20 [4]. They implemented on CVSI-2015 and ICDAR-2017 datasets which showed the highest overall accuracy than on SIW-13 and MLe2e datasets. CNN can be used in the multimedia area to recognize human activity from video frames [5]. Wang et al. in 2016 used some frames from the video and extract sequence features, but using CNN, the output is fed into long short-term memory (LSTM) model for video activity recognition. Jiang et al. in 2019 used two CNN algorithms parallelly [6] for multi-spectral RGB-NIR image classification. First, RGB-CNN algorithm is used in between R, G, B bands of aerial image for extracting features, and second, NIR-CNN algorithm is used to extract features from R, G, B and NIR bands of that image. Both features are fused in a fully connected layer and then classified. By using this method, RGB-NIR data are fully exploited. Lei et al. in 2019 used dilated CNN [5] instead of taking convoluted kernels for image classification. Secondly, they used hybrid dilated CNN which is a stack of dilated CNN having with different dilation rates, to reduce loss incurred in dilated CNN. As compared to traditional CNN [3, 4], this dilated CNN performs better with reduced training time by 12.99% and with the improvement of training accuracy by 2.86%. Secondly, with the HDC model, it reduces the training time to 2.02% and improves the accuracy of training by 14.15% and also testing by 15.35%. CNN can be used with other algorithm to detect a smile in human face. Qu et al. [7] compared two techniques of face detection algorithm such as face recognition using a color algorithm and AdaBoost Haar-like feature algorithm [8, 9]. After the face recognition, CNN is used for smile detection in that face. They also trained CNN with different optimizers [10]. CNN with RMSprop with momentum shows the highest accuracy of 92.09%, while CNN with Adam optimizer shows a maximum accuracy of 92.09%. But it shows only 69.53% with its original image. In the whole picture, the location of the face must be detected by using some face detection algorithm. Then CNN is applied to find special features (in reference [11], CNN was applied for a smile detection) depending upon your cause of research.

5 Conclusion

Deep learning provides a vast field, where image processing can be done with many available algorithms. Deep learning has practicability to solve many real-life problems. It has the ability of unsupervised learning with real-world datasets. So, CNN is one of the best algorithms of deep learning which performs better with the highest accuracy. CNN models are widely used by the researcher for many purposes. Some of them are discussed in this review.

References

1. Du, X., Cai, Y., Wang, S., Zhang, L.: Overview of deep learning. In: 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), pp. 159–164. Wuhan (2016)
2. Kido, S., Hirano, Y., Hashimoto, N.: Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN). In: International Workshop on Advanced Image Technology (IWAIT), pp. 1–4. Chiang Mai (2018)
3. Lei, X., Pan, H., Huang, X.: A dilated CNN model for image classification. IEEE Access **7**, 124087–124095 (2019)
4. Lu, L., Yi, Y., Huang, F., Wang, K., Wang, Q.: Integrating local CNN and global CNN for script identification in natural scene images. IEEE Access **7**, 52669–52679 (2019)
5. Wang, X., Gao, L., Song, J., Shen, H.: Beyond frame-level CNN: saliency-aware 3-D CNN with LSTM for video action recognition. IEEE Signal Process. Lett. **24**(4), 510–514 (2017)
6. Jiang, J., Feng, X., Liu, F., Xu, Y., Huang, H.: Multi-spectral RGB-NIR image classification using double-channel CNN. IEEE Access **7**, 20607–20613 (2019)
7. Qu, D., Huang, Z., Gao, Z., Zhao, Y., Zhao, X., Song, G.: An automatic system for smile recognition based on CNN and face detection. In: IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 243–247. Kuala Lumpur, Malaysia (2018)
8. Ma, L., Bai, L.: A face detection algorithm based on Adaboost and new haar-like feature. IEEE Chifeng University Chi/eng, Inner Mongolia Autonomous. China (2016)
9. Wan, L., Chen, P.: Face detection method based on skin color and AdaBoost algorithm. In: Fourth International Conference on Computational and Information Sciences. Chengdu, Sichuan China (2010)
10. Teow, M.Y.W.: Understanding convolutional neural networks using a minimal model for handwritten digit recognition. In: IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS), pp. 167–172. Kota Kinabalu (2017)

11. Han, M., Chen, J., Li, L., Chang, Y.: Visual hand gesture recognition with convolution neural network. In: 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), pp. 287–291. Shanghai, China (2016)

An Optimized Method for Arrhythmia Classification Using Artificial Neural Network

Sipra Sahoo, Saumendra Kumar Mohapatra, Bishnupriya Panda, Mihir Narayan Mohanty, and Sateesh Kumar Pradhan

Abstract Research based on biomedical data analysis is increasing day by day. Cardiac problem is one of major causes of death worldwide. Accurate classification and analysis of cardiac data is more essential to reduce the death rate. In this work, authors have classified cardiac data collected from UCI repository by using optimized model of artificial neural network (ANN). Weights are optimized using modified genetic algorithm (MGA) technique. Five different types of arrhythmia data are considered and classified. The results found about 90% accuracy. The obtained result is compared with some earlier works, and it is found that ANN-MGA is performing better.

Keywords ECG · Arrhythmia · ANN · Optimization · MOGA

1 Introduction

Electrocardiogram (ECG) is one of the important sources for the analysis of any kind of cardiac disease. Electrical impulses that occur during each heartbeat can be visualized in ECG. Accurate analysis of the ECG is most important in the diagnosis of heart disease. Any changes in a normal ECG can occur during various cardiac abnormalities like atrial fibrillation, ventricular tachycardia, myocardial infarction, hypokalemia, etc. Arrhythmia is a type of cardiac illness, and it can be detected by strongly analyzing waves in ECG [1]. It causes due to the abnormal electrical activity in the heart. In this case, the heartbeat rhythms fast or slow. The factors that

S. Sahoo · S. K. Mohapatra · B. Panda
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India

M. N. Mohanty (✉)
Department of Electronics and Communication Engineering, ITER, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: mihirmohanty@soa.ac.in

S. K. Pradhan
Department of Computer Science and Application, Utkal University, Bhubaneswar, Odisha, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_33

cause cardiac arrhythmia are smoking and alcohol consumption, mental stress, diet, diabetes, etc.

Due to the manual data analysis system, it is difficult to extract useful information from the clinical data. Computer-based automated disease diagnosis system will be most useful in medical sectors. The system will automatically take the decision by analyzing the data collected through a different medical test. Data includes ECG, ultrasonic images, and MRI which are useful for analysis and diagnosis. Machine learning-based medical data analysis is one of the advanced technologies that can reduce human interaction by enhancing the machine efficiency and will be cost-effective. For early detection of cardiac arrhythmia, a real-time automatic ECG analysis system will be the best support to the clinicians [2]. The characteristic of the ECG varies for different patients in different physical and temporal conditions. Due to these variations in ECG, the task is very difficult for analysis. This requires to develop an automatic ECG classification system [3]. For development of accurate ECG classifier, it is important to extract the useful features from data. Here, in the proposed work, an ANN-based classifier is designed with multiple offspring genetic algorithm (MOGA) optimization technique for the classification of five different types of cardiac arrhythmia.

In the course of the most recent thirty years, the genetic algorithm and their mixtures have been applied to different optimization problems, by reason of their various preferences: free subordinate attributes, straightforward readiness of the optimization model, the parallel idea of the pursuit, and so forth. One urgent issue for the genetic algorithm achievement, particularly for the troublesome issues, is to maintain a strategic distance from the untimely intermingling of the calculation to imperfect areas. The premature convergence of a genetic algorithm emerges when the qualities of some high evaluated genes rapidly achieve overwhelm the population, obliging it to merge to a neighborhood optimum. For this situation, the genetic operators cannot create further descendents better that the guardians the calculation capacity to proceed with the quest for better arrangements is in this way considerably diminished. To keep away from the premature convergence, a genetic algorithm is basic to safeguard the population decent variety during the development.

2 Related Literatures

Automated ECG classification can help the cardiologist for the diagnosis of any type of cardiac abnormalities. In the last few decades, several algorithms have been developed by the researchers for the automatic classification of the cardiac signal. Preprocessing, feature extraction, and classification are the three basic steps in ECG signal classification, and multiple methods have been applied by the researchers for each of these processes. Classification of normal and coronary artery disease was performed by applying higher-order statistics and spectra (HOS) method [4]. From each heartbeat, HOS bispectrum and cumulants features were extracted. For dimensionality reduction, authors have taken PCA. KNN and decision tree classifiers were used for the classification purpose as the performance with fewer features is suitable. Support vector machine (SVM) classifier was used by the authors for the detection of coronary artery disease [5]. They have used PCA for the feature selection purpose, and 79% classification accuracy was achieved by them. Genetic algorithm (GA) and binary particle swarm optimization (BPSO) methods were used for feature selection [6]. SVM with *k*-fold cross-validation was considered for classification purpose, and 81% classification accuracy was obtained from their proposed system. *S*-transform and wavelet transform were used for feature extraction [7]. Multilayer perceptron was considered for the classification of normal and abnormal ECG beats. A long-term ECG classification framework was presented in a study [8]. Exhaustive *k*-means clustering technique was used for getting an optimal number of key beat and master beats from the ECG waveform. Backpropagation algorithm-based classifier was introduced by the authors for the classification purpose. Average accuracy for their proposed system was 99.04%. In certain cases, multiple classifiers for ECG classification were used. Automatic detection of the coronary artery from the cardiac signal using four different classifiers was performed [9]. In the first stage, authors have applied DWT for decomposition purpose. Further, the dimensions of the wavelet coefficients were reduced by applying PCA, LDA, and ICA. After the selection of suitable features from the cardiac signal, the classification was performed by using KNN, SVM, PNN, and GMM. From their result, it has been observed that ICA with GMM classifier was giving better result as compared to the other three types of classifiers. Soft computing techniques have been utilized for ECG signal classification in different studies. Radial basis function neural network-based [10] and block-based neural network (BBNN) [11] were applied in some studies for automatic ECG classification. Similarly, the neural network was used to predict arrhythmia [12]. They have claimed the classification accuracy of 67%. Data mining method such as attribute selection and EM-based data clustering was introduced by authors in [13]. The dimension of the feature vector is reduced by correlation-based selection technique. A rule-based classifier was designed for classifying the ECG, and higher-order spectra (HOS) were introduced by the authors in [14]. For reduction of the dimension, ICA was considered and the classification was occurred by using multiple classifiers. From their result, KNN classifier was giving better classification accuracy as compared to other classifiers. Optimum path forest (OPF)-based classifier was chosen by the authors in [15].

In this paper, multiple offspring competition method is applied which is basically based on the reproduction process of genetic algorithm. In this method, multiple offsprings are generated generation by generation; then, competition takes place between the offsprings, and the winner of them became the real offspring. Three generations that are normal generation, strongly mutated generation, and queen bee generation strategies have been used for generating multiple offsprings. At the initial stage, winners generate offsprings and they have diversity [16]. Evolution speed can be accelerated due to queen bee generation phase, and strongly mutated generation helps the genetic algorithm to fall into local optimum. Objective of multiple offspring competition is to help GAs not to fall into local optimum which leads to premature convergence problem. This method is effective than the queen bee genetic algorithm as it does not need additional parameters and empirical selection of parameters.

3 Proposed Methodology

Five types of cardiac arrhythmia are classified using artificial neural network with multiple offspring genetic algorithm (MOGA) optimization. Cardiac data can be collected from the direct patient in hospitals or open source databases available in Internet. UCI Cleveland heart disease data is used in this work. The dataset contains 303 patients' samples, and each sample is characterized with 13 features. The details of the Cleveland database are presented in Table 1. 70% data is divided for training purpose, and the rest of the 30% data is then taken for test set.

Artificial neural network (ANN) is used for the classification purpose. Weights of the artificial neural network model are optimized using MOGA optimization technique.

Table 1 Cleveland heart disease dataset

Features	Explanation
Age	Patient age
Sex	1 represents male and 0 represents female patients
Cp	Different types of pain in checst (1 = typical angina, 2 = atypical angina, 3 = non-angial pain, 4 = asymptomatic)
Trestbps	Blood pressure of the patient at the admission time to hospital
Chol	Cholesterol of the patient in mg/dl
Fbs	Sugar in blood at fasting time
Restecg	ECG report (for normal it is 0, patients having ST-T wave abnormality it is 1 and left ventricular is 2
Thalch	Pulse rate
Exang	Exercise encouraged angina (for yes it is 1 and for no it is 0)
Oldpeak	ST depression induced by exercise ST segment
Slope	Peak exercise slope (up sloping = 1, flat = 2, down sloping = 3
Ca	Amount of main vessels (0–3) colored by fluoroscopy
Thal	3 shows the normal, fixed defect is denoted as 6 and reversible defect is denoted as 7

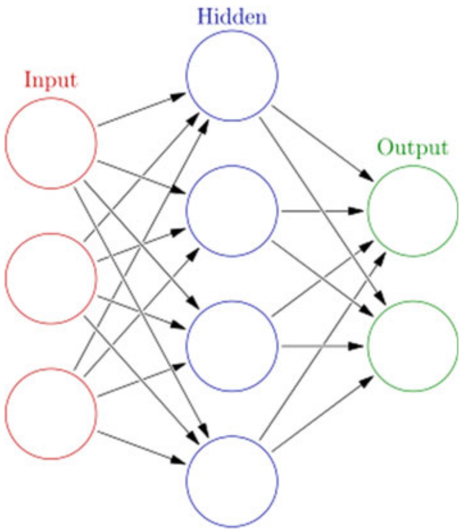


Fig. 1 Standard ANN structure

3.1 Artificial Neural Network (ANN)

An artificial neural network (ANN) is a generalized mathematical model which is based on biological nervous systems. The fundamental elements of neural networks are artificial neurons. Input, output, and hidden are three basic layers of a simple neural network as presented in Fig. 1. In feed-forward networks, the data flow is from input to output units, firmly in a feed-forward path. Both linear and nonlinear classification problems can be solved by applying ANN with various types of network structure and learning algorithm.

In neural network, the output of the hidden layer h and output layer y can be calculated as:

$$h = \sigma(W_1 x + b_1) \tag{1}$$

$$y = \sigma\left(W_2 h + b_2\right) \tag{2}$$

where W_1 and W_2 are the weights of neuron and x is the input. b_1 and b_2 are the bias. σ is the activation function. The weights are considered with the wavelet coefficients.

3.2 Multiple offspring Genetic Algorithm (MOGA)

The proposed genetic algorithm with multiple offspring competition is described in Algorithm 1. The asterisks in Algorithm 1 mean additional operations to the original GA. Each parent as shown in the algorithm generates three offspring and they compete with each other; finally, the winner offspring become real offspring for the parents.

1. t = 0
2. P(t) is initialized.
3. P(t) is evaluated
4. While (~ terminating condition)
5. Do
6. t = t + 1
7. Selection of P(t) from P(t – 1)
8. Recombination of P(t)
9. Normal generation of offspring
10. Do crossover with parents p1, p2
11. Do mutation with pm
12. Evaluate offspring o'
13. Strongly mutated generation of offspring (*)
14. Do crossover with parents p1, p2(*)
15. Do mutation with pm ((*)
16. Evaluate offspring o^q(*)
17. Queen bee generation of offspring (*)
18. Do crossover with parents p1, p2(*)
19. Do mutation with pm (*)
20. Evaluate offspring o^q (*)
21. Compete offsprings with fo, fo'& fogl(*)
22. Set the winner offsprings to the next offspring (*)
23. Evaluate P(t)
24. End

4 Experimental Setup and Result

The simulation is carried out by using MATLAB in a PC having i5 processor, 4 GB RAM, Windows operating system. Five different types of arrhythmias are classified in the proposed work. ANN classifier is designed with MOGA optimization. Detail description of the weight optimizer and ANN structure is presented in Table 2. The resultant weight from the trained dataset has been tested using sample test data. The performance is evaluated using MAE, MedAE, MSE, RMSE and is presented in Table 4.

In Table 3, the confusion matrix obtained by ANN with MOGA is presented. From this confusion matrix, around 90.08% classification result is archived and it is quite good as compared to the previous work presented in Tables 4 and 5.

Table 2 Parameters of the proposed model

ANN	MOGA
Size of the input node = 4 Total number of hidden layers = 1	Population size-100
Nodes in the hidden layers-5	Total number of iteration-100
Learning rate-0.125	$a = 2$–0
Total number of iteration-100	$A = [-1, 1]$

Table 3 Confusion matrix obtained using ANN-MOGA

Class	Normal	Coronary artery	Anterior myocardial infarction	Inferior myocardial infarction	Sinus tachycardy	Total	True %
Normal	**61**	4	1	0	0	66	
Coronary artery	1	**20**	1	0	0	22	
Anterior myocardial infarction	0	1	**14**	0	0	15	
Inferior myocardial infarction	0	2	1	**11**	0	14	
Sinus tachycardia	0	0	1	0	**3**	4	
Predicted	62	27	18	11	3	121	109/121 * 100 = 90.08264

Table 4 Performance measuring parameters

Parameters	Result
MAE	0.219667
MSE	0.049278
RMSE	0.221986

Table 5 Comparison of obtained result with earlier work

Reference	Method	Result (%)
Mustaqeem et al. [15]	Optimum path forest	68
De Chazal et al. [11]	LDA	76
Mustaqeem et al. [15]	KNN	76.6
Proposed work	ANN-MOGA	90.08

5 Conclusion

Cardiac problem is a widespread disease and is vital to detect accurately. Though numerous researchers have worked on it, still different techniques were developed to test the same. In this work, ANN classifier with MOGA is used for the classification purpose, and from this, we have achieved around 90% classification result which is quite good as compared to other works. Further accuracy can be improved by the use of different classification algorithms and other optimization techniques.

References

1. Mohapatra, S.K., Mohanty, M.N.: Analysis of resampling method for arrhythmia classification using random forest classifier with selected features. In: 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), pp. 495–499. IEEE, (Year)
2. Polat, K., Güneş, S.: Detection of ECG Arrhythmia using a differential expert system approach based on principal component analysis and least square support vector machine. Appl. Math. Comput. **186**, 898–906 (2007)
3. Hoekema, R., Uijen, G.J., Van Oosterom, A.: Geometrical aspects of the interindividual variability of multilead ECG recordings. IEEE Trans. Biomed. Eng. **48**, 551–559 (2001)
4. Acharya, U.R., Sudarshan, V.K., Koh, J.E., Martis, R.J., Tan, J.H., Oh, S.L., Muhammad, A., Hagiwara, Y., Mookiah, M.R.K., Chua, K.P.: Application of higher-order spectra for the characterization of coronary artery disease using electrocardiogram signals. Biomed. Signal Process. Control **31**, 31–43 (2017)
5. Babaoğlu, I., Fındık, O., Bayrak, M.: Effects of principle component analysis on assessment of coronary artery diseases using support vector machine. Expert Syst. Appl. **37**, 2182–2185 (2010)
6. Babaoglu, İ., Findik, O., Ülker, E.: A comparison of feature selection models utilizing binary particle swarm optimization and genetic algorithm in determining coronary artery disease using support vector machine. Expert Syst. Appl. **37**, 3177–3183 (2010)
7. Das, M.K., Ari, S.: ECG beats classification using mixture of features. Int. Sch. Res. Not. (2014)
8. Kiranyaz, S., Ince, T., Pulkkinen, J., Gabbouj, M.: Personalized long-term ECG classification: a systematic approach. Expert Syst. Appl. **38**, 3220–3226 (2011)
9. Giri, D., Acharya, U.R., Martis, R.J., Sree, S.V., Lim, T.-C., VI, T.A., Suri, J.S.: Automated diagnosis of coronary artery disease affected patients using LDA, PCA, ICA and discrete wavelet transform. Knowl. Based Syst. **37**, 274–282 (2013)
10. Lewenstein, K.: Radial basis function neural network approach for the diagnosis of coronary artery disease based on the standard electrocardiogram exercise test. Med. Biol. Eng. Compu. **39**, 362–367 (2001)
11. De Chazal, P., O'Dwyer, M., Reilly, R.B.: Automatic classification of heartbeats using ECG morphology and heartbeat interval features. IEEE Trans. Biomed. Eng. **51**, 1196–1206 (2004)
12. Abdou, A.-D., Ngom, N.F., Niang, O.: Classification and prediction of arrhythmias from electrocardiograms patterns based on empirical mode decomposition and neural network. In: International Conference on e-Infrastructure and e-Services for Developing Countries, pp. 174–184. Springer (Year)
13. Sufi, F., Khalil, I.: Diagnosis of cardiovascular abnormalities from compressed ECG: a data mining-based approach. IEEE Trans. Inf Technol. Biomed. **15**, 33–39 (2010)
14. Martis, R.J., Acharya, U.R., Min, L.C.: ECG beat classification using PCA, LDA, ICA and discrete wavelet transform. Biomed. Signal Process. Control **8**, 437–448 (2013)

15. Mustaqeem, A., Anwar, S.M., Majid, M., Khan, A.R.: Wrapper method for feature selection to classify cardiac arrhythmia. In: 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 3656–3659. IEEE (2017)
16. Wang, J., Ersoy, O.K., He, M., Wang, F.: Multi-offspring genetic algorithm and its application to the traveling salesman problem. Appl. Soft Comput. **43**, 415–423 (2016)

A Survey on Hybridized Gene Selection Strategies

Suman Kumari Panigrahi, Kaberi Das, Debahuti Mishra, and Bhaskar Kumar Veedhi

Abstract Due to high dimensionality of gene expression datasets, gene selection is an important step for improving gene expression data classification accuracy. This is true for the case of cancer classification using gene expression data. Gene selection is the identifying process for particular gene set from large no of genes. Gene selection means dimensionality reduction as the data dimensionality and storage requirements are reduced. In data mining, dimensionality is one of a major problem. It may affect the quality of learning model. One disadvantage may lead to have over-fitting to the learning model, and also, it requires high CPU time for testing the accuracy. So, to overcome such situation, traditional approaches (filter, wrapper, embedded and hybrid) are used for selecting genes. Hybrid method is the integration of two or more algorithms applied for gene selection. By using this method, we can able to overcome the disadvantages of other sole methods. From these studies, we got to know that hybrid method selects informative genes for classification and also improves the accuracy and run-time performance of the classifier. This paper describes various implementations of hybrid methods. This paper denotes review on few gene selection methods mainly those that have proposed in last few years.

Keywords Gene selection · Dimensionality reduction · Hybrid method · Gene expression

S. K. Panigrahi (✉) · K. Das · D. Mishra
Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, Odisha 751030, India
e-mail: sumankpanigrahi@gmail.com

K. Das
e-mail: kaberidas@soa.ac.in

D. Mishra
e-mail: debahutimishra@soa.ac.in

B. K. Veedhi
HCL Technologies Ltd, Noida, Uttar Pradesh, India
e-mail: BhaskarKumarV@hcl.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_34

1 Introduction

Genes are the information that is passed along in deoxyribonucleic acid (DNA). It tells the cells and molecules that how the organism should behave. Genes are responsible for all the characteristics we inherit. Mostly, microarray data contains thousand numbers of genes, in which most of the genes are irrelevant and redundant. Some of the genes carry noisy expression and are inappropriate; most of the genes are not able to detect the state of disease. Because of these, we choose to use gene selection method. Gene selection is the identifying process for particular gene set from larger number of genes. It reduces the data dimensionality and storage requirements.

Gene selection means data dimensionality reduction. Dimensionality is one of the major problems in most of the data mining task. It may affect the learned model and need high CPU time requirements for learning and testing the model. So, to overcome such situation, traditional approaches (filter, wrapper, embedded and hybrid) are used for selecting genes. Hybrid method is the integration of two or more algorithms applied for gene selection. By using this method, we can able to overcome the disadvantages of other sole methods. A hybrid method is able to select good genes for classification to improve run-time performance and accuracy of the classifier.

Rest of the paper is organized in a descriptive manner. In Sect. 2, methodologies of gene selection are discussed. Under Sect. 2, different traditional approaches are elaborated. Section 3 contains a survey on different hybrid methods used for gene selection, and Sect. 4 summarizes the study.

2 Methodology

2.1 Traditional Approach for Gene Selection

2.1.1 Filter Method

It is an open loop and the earliest method. Filter algorithm's main job is to measure the attribute characteristics. Here, the types of evaluation criteria are consistency, information, distance and dependencies are measured by filter algorithms. Most filter methods are univariate (Fig. 1).

Mishra and Sahu [1] analyze the two approaches of comparing the result from leukemia dataset. First approach is implementing *k*-means clustering and SNS ratio

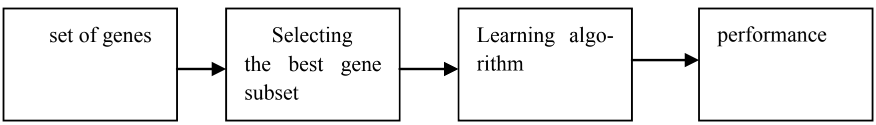


Fig. 1 Filter approach to gene selection

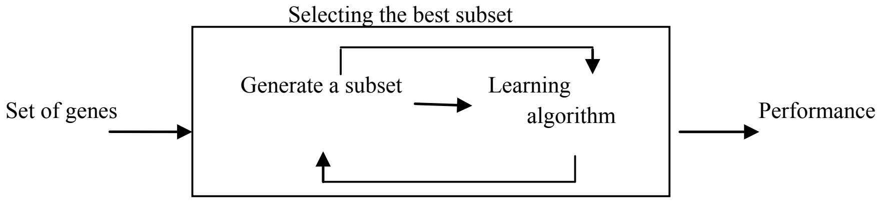


Fig. 2 Wrapper approach to gene selection

for gene ranking. Second approach is only of gene selection using signal-to-noise ratio. For validation, she has used *k*-nearest neighbor (*k*NN), support vector machine (SVM), feed forward neural network (*f*NN*)* and probabilistic neural network (PNN).

2.1.2 Wrapper Method

The second approach of gene selection is the wrapper method also known as closed loop method. The feature selection around the learning algorithm can be wrapped by this method. It utilizes classification error rate. And also utilizes the performance accuracy as feature evaluation criterion. It is better than open loop method. The maximal number of wrapper methods requires extensive computational resources since they are multivariate (Fig. 2).

Fukuta et al. [2] proposed a new method that is called LEAF method which stands for leave-one-out. It is a forward selection method for the analysis of gene expression data. He applied this method on three kinds of leukemia datasets, and those are ALL/AML, ALL/MLL and MLL/AML.

This type of gene selection method gives better classification accuracy and time consuming than the filter method. To reduce the time complexity, Wang et al. [3] explore the wrapper method with k-nearest neighbor (*k*NN) classifier. To speed up the gene selection process, he constructs and maintains a classifier distance matrix.

2.1.3 Embedded Method

This is the third approach of gene selection. Embedded means fixed, which is a built-in feature selection mechanism. It is used to fix the learning algorithm. To guide feature evaluation, it uses its properties. While maintaining similar performance, it is adequate and controllable than close loop method. It examines every attribute subset by skipping the tedious execution of classifiers (Fig. 3).

Bonilla-Huerta et al. [4] proposed a hybrid framework which is composed of two stages. In first stage, for preliminary gene selection, five traditional methods are combined. Then, by using embedded genetic algorithm (GA), tabu search (TS), support vector machine (SVM), different relevant gene subsets are selected. In this process, a gene subset is obtained which is consisting of the most apposite genes.

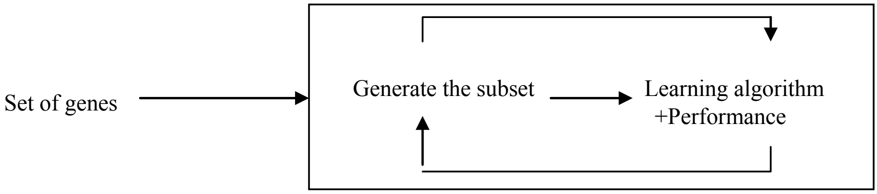


Fig. 3 Embedded approach to gene selection

Li et al. [5] purposed a conditional mutual information and also purposed an adaptive sparse group lasso. They proved that the result taken on four cancer datasets signifies that the purposed method can perform classification better and increase the accuracy of gene selection.

3 A Survey on Hybrid Methods

The combination of two different methods can be called as hybrid method, two methods are same criteria and feature selection approach. Filter and wrapper methods make the best combination and are the highly preferred hybrid method. As compared to filter method, performance and accuracy are higher. Computational result is also better in comparison with wrapper approach. This method has some disadvantages such as dependent in the combination of different gene selection methods.

By using DNA microarray technology, gene expression measured for every sample facing a great challenge. So, for cancer classification, Jain et al. [6] proposed a hybrid model integrating CFS and iBPSO. They have taken 11 different cancer type dataset. For 7 out of 11 dataset, their result is showing nearly 100% classification accuracy.

For many difficult problems, ant colony optimization (ACO) has been applied successfully. Ants can come across the best feature composition as they travel over the graph. So, based on ACO, Chen et al. [7] proposed a new rough set method for feature selection. This method adopts mutual information, and their algorithm can provide systematic solution to find minimal subset.

For small training sets, features may be harmful as it is harder to evaluate for its irrelevancy and redundancy. Large number of features lead to the high memory usage. So, to improve the accuracy and to remove the irrelevant features, FS is a solution. Here, Inbarani et al. [8] proposed a new supervised feature selection based on hybridization of particle swarm optimization (PSO), particle swarm optimization relative reduct (PSO-RR) and particle swarm optimization quick reduct (PSO-QR) for disease diagnosis. Their result proved the efficiency of this technique.

Sharbaf et al. [9] proposed an of filter method using fisher criterion. This approach reduces the time complexity and search space. Then, a wrapper method based on cellular learning automata (CLA) with ant colony optimization (ACO) to improve the classification accuracy. Their result can find the smallest subset of genes with maximum accuracy.

Xue et al. [10] proposed a new method where initial strategies and updating mechanisms are compared with the traditional strategies and traditional mechanisms. So that a new approach forms, i.e., PSO (4–2), which is compared with traditional FS method and PSO-based method. Their results are taken of twenty datasets which show the higher classification performance and reduce the computational time.

Apolloni et al. [11] gave new and efficient hybrid algorithms called as BDE-Xrank and BDE-Xrank1. Both algorithms are the combination of wrapper method and FS method based on BDE algorithm. Their result shows robustness and also gives highly accurate solution at a very earlier stage.

Sahu and Mishra [12] proposed a new feature selection approach which used signal-to-noise ratio (SNR) score and optimization technique PSO for high-dimensional cancer microarray data. This method is divided in two different stages. In first stage, *k*-means clustering is used, and for the ranking of each gene in cluster, SNR score is used. In the second stage, PSO is used for the input of new feature subset SVM, *k*NN and PNN are for evaluator, and for validation, cross validation approach is used.

In cancer diagnosis and treatment, an important problem is cancer classification, and the effective method is gene selection which increases the classification accuracy. Here, Motieghader et al. [13] proposed a new hybrid meta-heuristic algorithm. This algorithm is the combination of genetic algorithm and learning algorithm (GALA). They have taken six different cancer datasets for evaluation of the performance of GALA, and the experiment provided a remarkable result.

For classification purpose, Wang et al. [14] proposed a new bacterial colony optimization (BCO) method with multi-dimensional population (MDP). This BCO-MDP is basically presented for FS procedure. This method can be used to address the combinational trouble associated with FS. With regards to feature size and efficiency, this proposed method is illustrated to be higher to the binary algorithm. It lowers the computational complexity in comparison with the other popular algorithms.

One of the most major problems is “Dimensionality.” To reduce the poor classification accuracy and high CPU time requirement, Zorarpaci et al. [15] proposed a new hybrid method which is the combination of artificial BCO technique and DE algorithm. This method is for the purpose of feature selection of classification task. Their result is evaluated by 15 datasets. Their experimental result shows that the proposed hybrid method can able to select good feature to improve accuracy and run-time performance.

Lopez–Garcia et al. [16] proposed a new hybrid meta-heuristic approach. This method is the combination of genetic algorithm (GA) and cross entropy (CE). For solving continuous optimization function. They tested this experiment on twenty-four continuous benchmark functions. Their result shows that the best performing method with high dimensionality. On the combination of GA and PSO, a new FS method is proposed by Ghamisi and Benediktsson [17]. This new method can select informative features and performs better than other methods.

Mafarja and Mirjalili [18] proposed a new hybrid model for design different FS techniques. This hybridized method is the integration of whale optimization algorithm (WOA) and simulated annealing (SA). Their main goal of using simulated annealing is for searching the most favorable area located by whale optimization algorithm. Their result shows the improvement of classification accuracy and selects the most communicative attributes.

Lu et al. [19] proposed a new hybrid FS algorithm which is the integration of mutual information maximization (MIM) and adaptive genetic algorithm (AGA). Their experimental results reduce the dimension of gene data and reduce the redundancy. This new approach also gives high classification accuracy. They apply the four different classifiers to signify the robustness of this new MIMAGA algorithm.

Cancer classification still remains a problem on gene expression data as there is large number of genes present compared to available training datasets. So here, Chinnaswamy and Srinivasan [20] proposed a hybrid feature selection model that is the combination of cross entropy (CE) with particle swarm optimization (PSO). Their hybrid approach reduces the effective level of gene expression. As it acquires higher classification accuracy, it uses less number of features.

Hsu et al. [21] proposed a hybrid feature selection model which is the integration of two FS methods that are filter method and wrapper method. This hybrid method takes the advantage of both methods. This mechanism is scrutinized by two bioinformatics problems, and those are protein disordered region prediction and gene selection in microarray data.

Nowadays, meta-heuristic optimization techniques become popular for problem solving and give the near optimal solutions. Thangaraj et al. [22] proposed a new hybridized meta-heuristic approach which is the combination of two algorithms: particle swarm optimization (PSO) and genetic algorithm (GA). They made three classifications to get the positive results, and those are (a) hybridization of PSO and GA (b) hybridization of PSO and DE (c) hybridization PSO with local and global search methods.

Ali and Tawhid [23] proposed a new hybrid method of particle swarm optimization (PSO) and genetic algorithm (GA) for simplifying model of energy function. This proposed algorithm is naming as HPSOGA. Here, three mechanisms are carried out to solve the large-scale global optimization problems. HPSOGA is compared with standard PSO, and their experiment gives more promising result as compared to others.

In classification of microarray data, the main goal is to select small number of relevant genes to get the high classification accuracy. Bonilla–Huerta et al. [24] proposed a hybrid filter–wrapper method. They have used RMPX with GA in integration with LDA. Their experimental result improves the performance of gene selection. Elyasigomari et al. [25] proposed a method called two-stage gene selection method (MRMR-COA-HS). MRMR feature selection is used to select relevant gene subset. This proposed algorithm is significantly used least amount of genes and maintains high classification accuracy.

4 Summary

This paper provides a review on hybridized gene selection procedures. It narrates the DNA microarray, the problem persists in gene selection method including high dimensional, noisy and irrelevant data. It gives the details of traditional approaches (filter, wrapper, embedded, hybrid). These methods are used to read the best disease classification and high classification accuracy. Many researchers put their great efforts to examine the hybrid method which is the combination of two or more methods, and the advantages of this method prove to give appreciative results. As far as hybrid methods for gene selection influences the future success. Recently, the work has done by many researchers on hybrid method for gene selection reaches the best result in comparison with other.

References

1. Mishra, D., Sahu, B.: A signal-to-noise classification model for identification of differentially expressed genes from gene expression data. In: 2011 3rd International Conference on Electronics Computer Technology, vol. 2, pp. 204–208. IEEE (April, 2011)
2. Fukuta, K., Nagashima, T., Okada, Y.: Leaf: leave-one-out forward selection method for cancer classification using gene expression data. In: 2010 IEEE/ACIS 9th International Conference on Computer and Information Science, pp. 31–36. IEEE (August, 2010)
3. Wang, A., An, N., Chen, G., Li, L., Alterovitz, G.: Accelerating incremental wrapper based gene selection with *K*-Nearest-Neighbor. In: 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 21–23. IEEE (November, 2014)
4. Bonilla-Huerta, E., Hernandez-Montiel, A., Morales-Caporal, R., Arjona-López, M.: Hybrid framework using multiple-filters and an embedded approach for an efficient selection and classification of microarray data. IEEE/ACM Trans. Computat. Biol. Bioinf. (TCBB) **13**(1), 12–26 (2016)
5. Li, J., Dong, W., Meng, D.: Grouped gene selection of cancer via adaptive sparse group lasso based on conditional mutual information. IEEE/ACM Trans. Comput. Biol. Bioinf. **15**(6), 2028–2038 (2017)
6. Jain, I., Jain, V.K., Jain, R.: Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. Appl. Soft Comput. **62**, 203–215 (2018)
7. Chen, Y., Miao, D., Wang, R.: A rough set approach to feature selection based on ant colony optimization. Pattern Recogn. Lett. **31**(3), 226–233 (2010)
8. Inbarani, H.H., Azar, A.T., Jothi, G.: Supervised hybrid feature selection based on PSO and rough sets for medical diagnosis. Comput. Methods Programs Biomed. **113**(1), 175–185 (2014)
9. Sharbaf, F.V., Mosafer, S., Moattar, M.H.: A hybrid gene selection approach for microarray data classification using cellular learning automata and ant colony optimization. Genomics **107**(6), 231–238 (2016)
10. Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms. Appl. Soft Comput. **18**, 261–276 (2014)
11. Apolloni, J., Leguizamón, G., Alba, E.: Two hybrid wrapper-filter feature selection algorithms applied to high-dimensional microarray experiments. Appl. Soft Comput. **38**, 922–932 (2016)
12. Sahu, B., Mishra, D.: A novel feature selection algorithm using particle swarm optimization for cancer microarray data. Procedia Eng. **38**, 27–31 (2012)

13. Motieghader, H., Najafi, A., Sadeghi, B., Masoudi-Nejad, A.: A hybrid gene selection algorithm for microarray cancer classification using genetic algorithm and learning automata. Inf. Med. Unlocked **9**, 246–254 (2017)
14. Wang, H., Tan, L., Niu, B.: Feature selection for classification of microarray gene expression cancers using Bacterial Colony Optimization with multi-dimensional population. Swarm Evol. Comput. **48**, 172–181 (2019)
15. Zorarpacı, E., Özel, S.A.: A hybrid approach of differential evolution and artificial bee colony for feature selection. Expert Syst. Appl. **62**, 91–103 (2016)
16. Lopez-Garcia, P., Onieva, E., Osaba, E., Masegosa, A.D., Perallos, A.: GACE: a meta-heuristic based in the hybridization of genetic algorithms and cross Entropy methods for continuous optimization. Expert Syst. Appl. **55**, 508–519 (2016)
17. Ghamisi, P., Benediktsson, J.A.: Feature selection based on hybridization of genetic algorithm and particle swarm optimization. IEEE Geosci. Remote Sens. Lett. **12**(2), 309–313 (2014)
18. Mafarja, M.M., Mirjalili, S.: Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing **260**, 302–312 (2017)
19. Lu, H., Chen, J., Yan, K., Jin, Q., Xue, Y., Gao, Z.: A hybrid feature selection algorithm for gene expression data classification. Neurocomputing **256**, 56–62 (2017)
20. Chinnaswamy, A., Srinivasan, R.: Hybrid feature selection using correlation coefficient and particle swarm optimization on microarray gene expression data. In: Innovations in Bio-Inspired Computing and Applications, pp. 229–239. Springer, Cham (2016)
21. Hsu, H.H., Hsieh, C.W., Lu, M.D.: Hybrid feature selection by combining filters and wrappers. Expert Syst. Appl. **38**(7), 8144–8150 (2011)
22. Thangaraj, R., Pant, M., Abraham, A., Bouvry, P.: Particle swarm optimization: hybridization perspectives and experimental illustrations. Appl. Math. Comput. **217**(12), 5208–5226 (2011)
23. Ali, A.F., Tawhid, M.A.: A hybrid particle swarm optimization and genetic algorithm with population partitioning for large scale optimization problems. Ain Shams Eng. J. **8**(2), 191–206 (2017)
24. Bonilla-Huerta, E., Duval, B., Hernández, J.C.H., Hao, J.K., Morales-Caporal, R.: Hybrid filter-wrapper with a specialized random multi-parent crossover operator for gene selection and classification problems. In: International Conference on Intelligent Computing, pp. 453–461. Springer, Berlin, Heidelberg (August, 2011)
25. Elyasigomari, V., Lee, D.A., Screen, H.R., Shaheed, M.H.: Development of a two-stage gene selection method that incorporates a novel hybrid approach using the cuckoo optimization algorithm and harmony search for cancer classification. J. Biomed. Inform. **67**, 11–20 (2017)

Indian Stock Market Prediction Based on Rough Set and Support Vector Machine Approach

Rudra Kalyan Nayak, Ramamani Tripathy, Debahuti Mishra, Vijay Kumar Burugari, Prabha Selvaraj, Abhisek Sethy, and Biswaranjan Jena

Abstract In recent trends, the stock market prediction has been treated as a challenging task for every people who have been associated with the financial market. Forecasting is the activity of historic data to set up the direction of future trends. It is a very difficult task to predict stock prices because, in every second, the market price is fluctuating. Most of the investors have an interest in doing research on prediction of price financial products such as gold, mutual fund, crude oil, currency exchange, and minerals using varieties of machine learning and data mining techniques. On the basis of different parameters of data like opening price, closing price, date, high and low, stock market price has been predicted. Basically for Indian markets, the two indices

R. K. Nayak (✉) · V. K. Burugari · A. Sethy
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation (Deemed to be University), Guntur, Andhra Pradesh, India
e-mail: rudrakalyannayak@gmail.com

V. K. Burugari
e-mail: vijaybru@gmail.com

A. Sethy
e-mail: abhisek052@gmail.com

R. Tripathy
Department of Master of Computer Application, United School of Business Management, Bhubaneswar, Odisha, India
e-mail: ramatripathy1978@gmail.com

D. Mishra
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, Odisha, India
e-mail: mishradebahuti@gmail.com

P. Selvaraj
Department of Computer Science and Engineering, VIT-AP University, Amaravati, Andhra Pradesh, India
e-mail: prabha.dw@gmail.com

B. Jena
Tata Consultancy Services, Bhubaneswar, Odisha, India
e-mail: biswarjena@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*, Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_35

such as Sensex for BSE (Bombay Stock Exchange) and nifty for NSE (National Stock Exchange) are the benchmark indices used for forecasting the market price. Day by day huge volumes of people are paying their keen interest in money trading and they want to become profit makers instantly. Here we proposed a novel rough set—Support Vector Machine (R-SVM) approach to predict the Indian stock market data. The R-SVM method is found to be prominent when compared with rough set based algorithms such as decision tree, Naïve Bayes, and artificial neural network in terms of prediction accuracy and complexity.

Keywords BSE · NSE · Rough set · SVM · Min-max · Principal component analysis

1 Introduction

For Indian scenario financial market analysis has been treated as the most important issue from past decades to till date. Financial market is nothing but trading of financial assets such as bonds, debentures, and shares between seller and buyer online. It also plays a vital role in allocating a variety of resources in our world economy. As it is a demanding challenge for all countries, so a lot of investors and researchers are willing to do their research for forecasting different financial products. Among all market products, stock price prediction takes a significant role in current days [1–5]. The primary purpose of the financial stock market is to regulate the exchange of stocks, as well as other financial assets and creating an environment through which all company stakeholders without side investors will get benefit in long run. Forecasting of stock market is a very tedious task for all researchers due to market price fluctuation in each time. In order to make graceful investment decisions to earn the best profits, market research is required. For this purpose, most of the investors have been implemented so many machine learning and data mining techniques for accurate forecasting of stock price. Most researchers use Artificial Neural Network (ANN), Support Vector Machine (SVM), and *K*-Nearest Neighbor (KNN), Fuzzy Cognitive Maps (FCM), and Logistic Regression Analysis for forecasting stock market price [1–10]. Therefore, we have proposed a hybrid model that is rough set with SVM for prediction of market price and found out the optimal solution which is benefited to all recent investors.

This script is prepared as follows: In part 2 related works are analyzed. Section 3 describes the dataset and methodology used and in part 4 experimental evaluation with output is depicted. Finally in part 5 conclusion of this paper is illustrated.

2 Related Works

Generally, stock market prediction has been the most intriguing task since so many years. As stock data is vacillating in nature and its prediction can make the investors aware of whether to invest in the financial market to get benefited, the researchers became very much concerned for its prediction. Different statistical, machine learning, and various techniques were used for stock market prediction. Some of the literatures are depicted here.

Xiaozhong et al. [11], in their paper, discussed financial stock market prediction on daily basis using dimensionality reduction. There they used ten years of data of daily direction of the S&P 500 Index ETF (SPY) return having 60 financial and economic features. The authors concluded that hybridizing the ANNs approach with the PCA provides better results among others. Furthermore, Kim et al. [12] have proposed a market trading system based on the dynamic time warping algorithm using KOSPI 200 index futures time-series data. The researchers investigated how their algorithm performs well with different parameters. Selvathmuu*et al.* [13] presented an overview of the prediction of stock market on tick data based on artificial neural networks. The researchers were discussing that it is not easy to predict this data due to its characteristics with lively nature. Therefore, they used neural network on basis of three distinct learning algorithms for stock market price forecasting in the Indian environment and also from this investigation authors were concluding that neural network approach provides better accuracy like 99.9% in comparison among other three approaches such as 96.2, 97, 98.9% consecutively. In 2015, author Lakshman Naik et al. [14] have proposed *k*-means clustering using MapReduce for BSE data prediction in the Indian market. In this case, researchers discussed all prediction techniques which are useful to the investors in the long term basis as it is capable to predict the company's next bid exactly based on the other industries that have alike trend with it. Patel et al. [15] discussed varieties of techniques like Moving Average and Neural Network and also this work focuses on a relative study on these approaches and finding the best techniques which provided a better performance on stock data.

3 Datasets with Methods Description

In the current manuscript, we have taken CNX nifty and BSE Sensex for predicting stock market in recent years. As we know both data sets are obtaining chief responsibilities on stock prediction in our country. NSE is treated as biggest stock exchange whereas BSE is known as an older version of stock exchange. Almost all trading in the Indian market depends on NSE and BSE exchange. These two most well-known indexes for stock market prediction for the Indian scenario are used now a day.

BSE is the older version of market index utilizes for equities and it was invented in 1986 which makes time-series statistics merely available from 1979 April to till

Table 1 Depiction of both data set (BSE sensex & CNX nifty)

Datasets	Whole number of sample data	Data series	Sample data for training	Sample data for testing
BSE sensex	2385	01-01-2009 to 10-10-2019	600	200
CNX nifty	2141	01-01-2010 to 10-10-2019	900	300

date. But in case of CNX nifty, it was created in 1990 and it also makes time-series statistics available from 1990 July to till date [16, 17]. Here we have taken data in the range given in the table. Table 1 describes the collected sample data and its range.

3.1 Proposed Model

Figure 1 depicts the working procedure of our model. In the first case, we retrieved aforesaid stock exchange data. Second, data set is smoothed using technical indicators [18, 19] and subsequently we have implemented min-max technology [18, 19] for scaling the data set. After that rough-decision tree [20, 21], rough-Naïve Bayesian [22] and rough-ANN [23] method afterward rough set with support vector machine approach has been employed over the smoothed dataset and results are analyzed to exhibit the proficiency of proposed model.

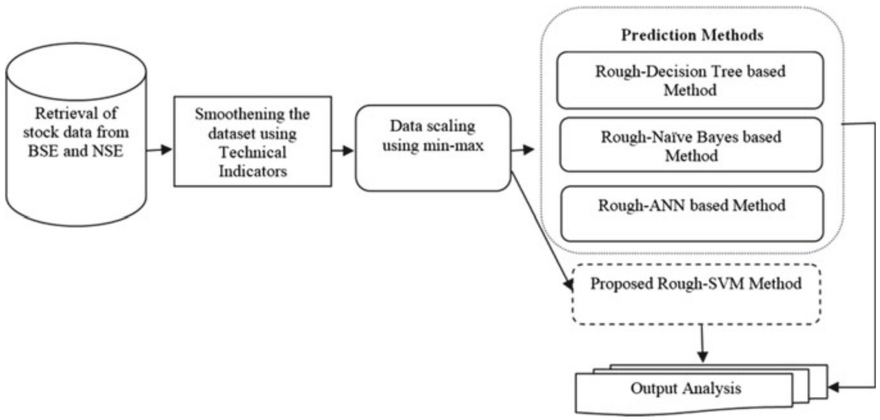


Fig. 1 Schematic layout of the proposed model

3.2 Description of Rough Set Algorithm

In the year 1980 rough set theory [24–29] was developed by Zdzislaw Pawlak. This theory is a simple mathematical tool that deals with inconsistent, vague, and imprecise knowledge. In current days, most of the researchers have focused their research on artificial intelligence and soft computing with data mining techniques for solving larger problems which are arising from diversified domains. In case of rough set theory, vagueness can be articulated by a boundary label of a set. Generally, it calculates the upper and lower label data set. It is mostly used for data reduction, decision rule generation, feature selection, pattern extraction, and feature extraction.

The theory of rough set is illustrated as $P = < U, V, S, R >$ by Z. Pawlak, with $S = E \cup F$, at which U is treated as universe of non-void group of items and also S is represented as non-void group of properties, E and F both subsets are recognized as decision feature collection and condition feature collection. $V = \bigcup_{d \in S} Vd$, everywhere V_d is symbolized as a collection of feature values of d, as well as the cardinality of $(V_d) > 1$ with $f\colon S \rightarrow V$ is known as characterization mapping.

Indiscernible relation: This relation is necessary for a well-known feature collection of each subclass $A \subseteq S$, an imperceptible relationship imp(A), that could be described in Eq. (1),

$$\mathrm{imp}(A) = \{(g, h) | (g, h) \in U^2, \forall_{c \in A}(c(g) = c(h))\} \tag{1}$$

Hence the concept of similarity relationship is not present here but an imperceptible relationship is taken into account. $[h]_{\mathrm{imp}(A)}$ or $[h]_A$ and $[h]$ pass on as equality family of an item if there is no confusedness. Plus after that $(U, [h]_{\mathrm{imp}(A)})$ pair is treated as an estimate space.

Lower and Upper bound sets: For a specified system $P = < U, V, S, R >$, taking into consideration $Z \subseteq U$ wherever Z is a subset, upper and lower bound sets [27] correspondingly be mentioned in Eqs. (2) and (3) with

$$\overline{\mathrm{appr}}(Z) = \{h \in U | [h] \cap Z \neq \phi\}, \tag{2}$$

$$\underline{\mathrm{appr}}(Z) = \{h \in U | [h] \subseteq Z\}, \tag{3}$$

Wherever $[h]$ is represented as equality family of h.

Therefore gathering of all equality family units is accepted as the ratio collection group of U, in addition to $U/S = \{[h] | h \in U\}$ denotes it. Then space of discussion is broken down into three disjoint parts, for instance, the positive, boundary, and negative [24–28] are as shown in Eqs. (4), (5), and (6), respectively.

$$M(Z) = \underline{\mathrm{appr}}(Z), \tag{4}$$

$$A(Z) = \overline{\mathrm{appr}}(Z) - \underline{\mathrm{appr}}(Z), \tag{5}$$

$$Q(Z) = U - \overline{\text{appr}}(Z), \tag{6}$$

If an object $h \in M\,(Z)$, as a result, it is a part of goal set Z positively. If $h \in A\,(Z)$, as a result, it would not be a part of target set Z positively. If $h \in Q\,(Z)$, as a result, it is difficult to make a decision whether h would be an element of target set Z.

3.3 Support Vector Machine

As theory suggests, support vector machine [30–33] is nothing but a well-known supervised learning approach that is mostly used for solving classification and regression problems. The main objective of this algorithm is that it creates a line or a hyper-plane which splits the data into classes. More vividly we can say that plot every data item as a point in n-dimensional space with the value of each feature being the value of a particular coordinate. After that, we do classification by finding the hyper-plane to distinguish the two classes very well. This algorithm is basically used in time-series platforms.

A training set which contains label pairs $(v_i, x_i), i = 1, \ldots, n$ anywhere $v_i \in \Re^n$ and $x \in \{\text{profit, loss}\}^i$, support vector machine necessitates result using optimization problem which is mentioned below.

$$\min_{z,c,\xi} \frac{1}{2} z^T z + B \sum\nolimits_{i=1}^{n} \xi_i \tag{7}$$

$$\text{Subject to} : x_i (z^T \Phi(v_i) + c \geq 1 - \xi_i, \xi_i \geq 0. \tag{8}$$

Decision function are given in Eq. (9)

$$q = pgn\left(\sum\nolimits_{i=1}^{n} y_i \alpha_i K(y_i, y) + \rho\right) \tag{9}$$

By the help of kernel function ϕ the training vector v_i is mapped into higher dimensional space. The separating surface depends on a subset of the original data known as a set of support vectors. $B > 0$ is the penalty parameter in case of error. According to the concept of SVM, data points are classified in hyper-planes basis but it becomes impracticable for finding linear solutions in two-dimensional spaces. So we eradicate this challenge by using kernel function $m(v_i, v_j) \equiv \Phi(v_i)^T \Phi(v_j)$ for multidimensional data. Using dissimilar kernels function support vector machine is trained which is shown in Eqs. (10) and (11) and in Eq. (12).

(a) Linear kernel : $(v_i, v_j) = v_i^T v_j$ (10)

(b) Polynomial kernel : $m(v_i, v_j) = (\gamma v_i^T v_j + r)^e, \gamma > 0,$ and (11)

(c) Radial Basis kernel(RBF) : $\left(v_i, v_j\right) = \exp\left(-\gamma \left\| v_i - v_j \right\|^2\right), \gamma$ (12)

On the basis of dataset all kernel functions such as B, γ, r, and e are initialized. Depending upon the training data size kernel parameters are affected. SVM theory tells that it can be both nonlinearly and linearly separable. Normally Kernel function selection depends on the dataset.

4 Experimental Evaluations

In this part, we have a wide-ranging experimental discussion on the proficiency and worth of our technique with the use of BSE Sensex and NSE nifty stock exchange data. In this manuscript, experiments are done taking Intel i3 processor, 4 GB RAM with 1 TB Hard disk. Here we have utilized windows 7 operating system and the entire code is written using python 3.

Basically, our investigation is organized in five stages.

Stage 1: *Collection of Dataset*: Retrieval of data is a very primary part of every work. Each proposal commonly deals with accurate dataset selection. In our case, we have collected stock indices: Sensex and nifty from BSE and NSE, respectively, for the Indian environment. The number of sensex data selected is 2385 in the range from 01-01-2009 to 10-10-2019 and 2141 number of nifty data are collected in the range from 01-01-2010 to 10-10-2019. Our data and prediction horizons are 1 week, 1 day, and 1 month. From BSE, the number of closing price samples for training and testing is taken as 600 and 200, respectively. Similarly from NSE, the number of closing price samples for training and testing is taken as 900 and 300 respectively.

Stage2: *Smoothening and Regeneration of dataset:* Data set is smoothened and regenerated using technical indicators [18, 19, 24]. For neat visualization of analysis graph of actual versus predicted data, it needs smoothening. For this purpose, we use some of the selected technical indicators such as simple moving average (SMA), Williams %*R*, true strength index (TSI), and relative strength index (RSI) [18, 19] given in Eqs. (13), (14), (15), and (16), respectively.

$$\mathrm{SMA}_i = \frac{1}{2M+1}(\mathrm{MA}(i+M) + \mathrm{MA}(i+M-1) + \cdots + \mathrm{MA}(i-M)) \quad (13)$$

$$\%R = \frac{\mathrm{high}_{M\mathrm{days}} - \mathrm{close}_{\mathrm{today}}}{\mathrm{high}_{M\mathrm{days}} - \mathrm{low}_{M\mathrm{days}}} \times -100 \quad (14)$$

$$\mathrm{TSI}(c_0, p, q) = 100 \times \frac{\mathrm{EMA}(\mathrm{EMA}(m, p), q)}{\mathrm{EMA}(\mathrm{EMA}(|m|, p), q)} \quad (15)$$

$$\mathrm{RSI} = 100 - \frac{100}{1 + \frac{\mathrm{EMA}(U,m)}{\mathrm{EMA}(D,m)}} \quad (16)$$

Stage 3: *Scaling of Smoothed Dataset*: In next phase, we have implemented min-max technology for scaling the data set mentioned in Eq. (17).

$$\tilde{n}^{ij} = \frac{n_{ij} - n_{\min_j}}{n_{\max_j} - n_{\min_j}} \quad (17)$$

where, n_{ij} is the jth attribute value of ith feature n_i, for the dataset, jth minimum value is $n_{\min_j}$ and maximum value is $n_{\max_j}$ and normalized price of ith day is $\tilde{n}^i$.

Stage 4: *Employment of Methods*: After getting all correct format data, we have applied rough-decision tree [20, 21], rough-Naïve Bayesian [22] and rough-ANN [23] based methods to detect the result. Later rough set with support vector machine approach is employed over the smoothed dataset. Rough set classifies *loss*, *profit,* and *no change* classes as lower approximation, upper approximation, and boundary regions in a promised way. Furthermore, SVM being a better classifier works effectively on the large dimension of nonlinear dataset. It can work well for small training dataset also. The regularization parameter C and kernel coefficient γ is chosen in the range $\{2^{-2}–2^{-3}\}$ and $\{2^{-2}–2^{-13}\}$, respectively, by tuning process in our work. The kernel function is chosen as Radial Basis Function [18] as per the requirement of dataset. Table 2 depicts the prediction accuracy of closing prices of different methods compared to our proposed method.

Stage 5: *Performance evaluation*: The performance is observed taking root mean square technique into account. Convergence speed of the proposed rough-SVM method is shown in Fig. 2.

Table 2 Performance investigation of rough-SVM approach and three afore-mentioned methods in percentage for prediction of loss or profit for daily, weekly, and monthly horizon

Method	Datasets	Time horizon		
		1 day	1 week	1 month
Proposed rough-SVM method	BSE sensex	89.05	87.88	84.73
	CNX nifty	88.01	86.5	84
Rough-decision tree method	BSE sensex	88.22	86.32	84
	CNX nifty	87.14	85.3	83.7
Rough-Naïve Bayes method	BSE sensex	83.51	81.4	80.2
	CNX nifty	82.6	80.33	80
Rough-ANN method	BSE sensex	82.23	81.54	80.11
	CNX nifty	82	81.62	80.1

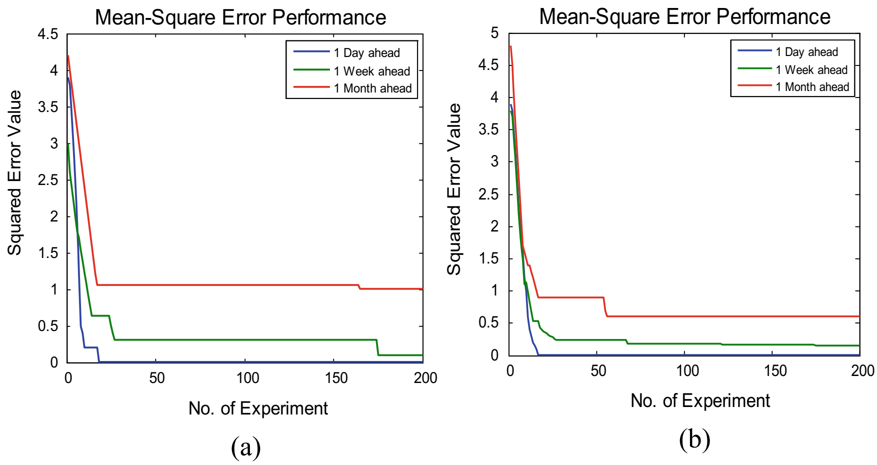


Fig. 2 Rate of convergence of rough-SVM method using mean-squared error: **a** BSE Sensex, **b** CNX nifty for daily, weekly, and monthly prediction

5 Conclusion

According to the Indian economy stock market prediction plays a vital role. It is a very difficult task for all researchers to forecast stock price due to variation of the market price in every second. So to determine price inflation in the market, analysts have been exploring various machine learning techniques. In this paper, we have implemented proposed rough with SVM algorithm for BSE Sensex and NSE nifty data set in Indian framework as rough set handles lower approximation, upper approximation, and boundary regions data as well. Further SVM is capable enough to classify large dimension nonlinear dataset in an effective way. Here we analyzed three methods such as decision tree, Naïve Bayes, and ANN embedded by rough set and compared with our proposed rough-SVM model. In this work, we have attained comparably decent prediction results than the above three methods and we say rough-SVM method outperforms the aforesaid methods in terms of prediction accuracy. This effort can be extended to embedment of some optimization methods to the proposed model for further finer results to reduce prediction error if possible.

References

1. Shen, S., Jiang, H., Zhang, T.: Stock market forecasting using machine learning algorithms. pp. 1–5, Department of Electrical Engineering, Stanford University, Stanford, CA (2012)
2. Sharma, A., Bhuriya, D., Singh, U.: Survey of stock market prediction using machine learning approach. In: 2017 International Conference of Electronics, Communication and Aerospace Technology (ICECA), vol. 2, IEEE (2017). https://doi.org/10.1109/iceca.2017.8212715

3. Loke, K. S.: Impact of financial ratios and technical analysis on stock price prediction using random forests. In: 2017 International Conference on Computer and Drone Applications (IConDA), IEEE (2017). https://doi.org/10.1109/iconda.2017.8270396
4. Zhang, X., Qu, S., Huang, J., Fang, B., Yu, P.: Stock market prediction via multi-source multiple instance learning, pp. 50720–50728, IEEE Access 6 (2018). https://doi.org/10.1109/access.2018.2869735
5. Kanade, V., Devikar, B., Phadatare, S., Munde, P., Sonone, S.: Stock market prediction: Using historical data analysis. International Journal. **7**(1) (2017). https://doi.org/10.23956/ijarcsse/v711/0112
6. Ahamed, A.T.M.S., Mahmood, N.T., Rahman, R.M.: An intelligent system to predict academic performance based on different factors during adolescence. J. Inf. Telecommun. **1**(2), 155–175 (2017). https://doi.org/10.1080/24751839.2017.1323488
7. Nayak, R. K., Kuhoo, Mishra, D., Rath, A. K., Tripathy, R.:. A novel look back N feature approach towards prediction of crude oil price. Int. J. Eng. & Technol. (UAE). **7**(3.34), 459–465. https://doi.org/10.14419/ijet.v7i3.34.19360
8. Grigoryan, H.: A stock market prediction method based on support vector machines (SVM) and independent component analysis (ICA). Database Syst. J. **7**(1), 12–21 (2016)
9. RautSushrut, D., Shinde, I.U., Malathi, D.: Machine learning approach in stock market prediction. Int. J. Pure Appl. Math. **115**(8), 71–77 (2017)
10. Lijuan, C., Tay, F.E.H.: Financial forecasting using support vector machines. Neural Comput. Appl. **10**(2), 184–192 (2001)
11. Xiao, Z., Enke, D.: Forecasting daily stock market return using dimensionality reduction. Expert Syst. Appl. **67**, 126–139 (2017). https://doi.org/10.1016/j.eswa.2016.09.027
12. Kim, S., Lee, H., Ko, H., Jeong, S., Byun, H., Oh, K.: Pattern matching trading system based on the dynamic time warping algorithm. Sustainability **10**(12), 4641 (2018). https://doi.org/10.3390/su10124641
13. Selvathmuu, D., Kumar, V., Mishra, A.: Indian stock market prediction using artificial neural networks on tick data. Financ. Innov. **5**(1), 16 (2019). doi: https://doi.org/10.1186/s40854-019-0131-7
14. Naik, R. L., Manjula, B.: Prediction of BSE stock data using mapreduce *K*-mean cluster algorithm. (2015)
15. Patel, H., Parikh, S.: Comparative analysis of different statistical and neural network based forecasting tools for prediction of stock data. In: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, ACM (2016). https://doi.org/10.1145/2905055.2905186
16. https://www.bseindia.com/indices/IndexArchiveData.html
17. http://www.nseindia.com/content/indices/ind_cnx_nifty.pdf
18. Nayak, R.K., Mishra, D., Rath, A.K.: A Naïve SVM-KNN based stock market trend reversal analysis for Indian benchmark indices. Appl. Soft Comput. **35**, 670–680 (2015). https://doi.org/10.1016/j.asoc.2015.06.040
19. Nayak, R.K., Mishra, D., Rath, A.K.: An optimized SVM-k-NN currency exchange forecasting model for Indian currency market. Neural Comput. Appl. **31**(7), 2995–3021 (2019). https://doi.org/10.1007/s00521-017-3248-5
20. Nair, B. B., Mohandas, V. P., Sakthivel, N. R.: A decision tree—Rough set hybrid system for stock market trend prediction. Int. J. Comput. Appl. **6**(9), 1–6 (2010). https://doi.org/10.5120/1106-1449
21. Wang, J.L., Chan, S.H.: Stock market trading rule discovery using two-layer bias decision tree. Expert Syst. Appl. **30**(4), 605–611 (2006). https://doi.org/10.1016/j.eswa.2005.07.006
22. Phichhang, O., Wang, H.: Prediction of stock market index movement by ten data mining techniques. Mod. Appl. Sci. **3**(12), 28–42 (2009)
23. Kyoung-jae, K., Lee, W.B.: Stock market prediction using artificial neural networks with optimal feature transformation. Neural Comput. Appl. **13**(3), 255–260 (2004). https://doi.org/10.1007/s00521-004-0428-x

24. Magda, B. F., El-Boghdadi, H. M., Omran, S. M.: Multi-objective optimization of technical stock market indicators using gas. Int. J. Comput. Appl. **68**(20) (2013)
25. Pawlak, Z.: Rough sets. Int. J. Parallel Prog. **11**(5), 341–356 (1982). https://doi.org/10.1007/BF01001956
26. Polkowski, L.: Rough sets. Physica-Verlag, Heidelberg (2002)
27. Pawlak, Z., Skowron, A.: Rough membership functions. In: Advances in the Dempster-Shafer theory of evidence, pp. 251–271. Wiley (1994)
28. Andrzej, S., Komorowski, J., Pawlak, Z., Polkowski, L.: Rough sets perspective on data and knowledge. In: Handbook of data mining and knowledge discovery, pp. 134–149. Oxford University Press, Inc. (2002)
29. Lech, P., Skowron, A.: Rough mereological calculi of granules: A rough set approach to computation. Comput. Intell. **17**(3), 472–492 (2001). https://doi.org/10.1111/0824-7935.00159
30. Corinna, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995). https://doi.org/10.1007/BF00994018
31. Francis EH. T., Cao, L.: Application of support vector machines in financial time series forecasting. Omega **29**(4), 309–317 (2001). doi: https://doi.org/10.1016/S0305-0483(01)00026-3
32. Chih-Wei, H., Lin, C. J.: A comparison of methods for multiclass support vector machines. IEEE Trans. Neural Netw. **13**(2), 415–425 (2002). doi: S1045-9227(02)01805-2
33. Upadhyay, V. P., Panwar, S., Merugu, R., Panchariya, R.: Forecasting stock market movements using various kernel functions in support vector machine. In: Proceedings of the International Conference on Advances in Information Communication Technology & Computing, p. 107, ACM (2016). https://doi.org/10.1145/2979779.2979886

Epileptic Seizure Detection Using DWT- and Rule-Based Variants of TSVM with KNN

Sumant Kumar Mohapatra and Srikanta Patnaik

Abstract The objective of our work is to develop an accurate automated seizure detection model that objectively evaluates the improvement on epileptic patients. This work used the Bonn University datasets for experimental works. Secondly, three types of features are extracted from EEG signals. Thirdly, all the features are combined. Next, all the features sets were inputted to the rule-based twin support vector machines ($TSVM_s$) to detect normal, ictal, and pre-ictal EEG segments. The developed seizure detection DWT-KWMTSVM method achieved excellent performance with the average sensitivity, specificity, accuracy, and positive predictive value; Matthews correlation coefficients are 99.5%, 96.96%, 99.26%, 99.57%, and 96.34%, respectively.

Keywords EEG signal · Epileptic seizure · DWT · Rule-based TSVMs · KNN

1 Introduction

Epilepsy detects the disorder condition of the mental function with the study of the irregular EEG signal flow to the human brain [1]. According to 2009 report of WHO, about 40–50 million people are affected by epileptic disorder diseases across the world [2, 3]. EEG signals are recorded with the help of electrodes that are placed on the various affected area on the human brain, analyze, and evaluate the condition of epilepsy [4–6]. Recording of the EEG signals is accomplished by placing the electrodes on the scalp with the help of 10–20 electrode placement systems. Electrodes are placed with some specified areas of the brain; these are frontal lobe, temporal lobe, etc., and the placement schemes are discussed in more detail at [7]. In [8], Kumar et al. presented a seizure detection method using DWT-based ApEn

S. K. Mohapatra (✉) · S. Patnaik
Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha, India
e-mail: sumsusmeera@gmail.com

S. Patnaik
e-mail: srikantapatnaik@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_36

and artificial neural network (ANN). The authors generated AnEn feature for classification. They achieved 94% accuracy when studied a Bonn University database. In [9], authors detected seizure and achieved 96.28% of accuracy from their proposed model. Later on, Orhan et al. in [10] designed an EEG detector using the *K*-means clustering and a MLPNN classifier.

2 Proposed Method

2.1 Clinical Dataset

For this research work, EEG datasets by the Department of Epileptology, University of Bonn, Germany [11] are used.

2.2 Feature Extraction

In the present work, three types of features are extracted from the recorded EEG signals. The time domain features like mean, median, mode, variance, standard deviation, skewness, and Kurtosis are extracted by using different methods [12, 13]. The frequency domain features such as maximum, minimum, mean, and standard deviation of the wavelet coefficients are extracted from each sub-band. There are three entropy-based features used in this study as mentioned in Eqs. (1)–(3).

$$\text{Sample} - \text{Entropy}, \ \text{SE}_n(k, r.N) = \ln\left(\frac{A(k)}{B(k-1)}\right) \tag{1}$$

$$\text{Phase} - \text{Entropy}, \ \text{PE}_n S1 = -\sum\nolimits_k P_k \log P_k \tag{2}$$

$$\text{Spectral} - \text{Entropy}, \ \text{SE}_n = \sum\nolimits_f P_f \ \log\left(\frac{1}{P_f}\right) \tag{3}$$

2.3 Rule-Based Twin SVM Classifiers

In this research work, for classification of normal, interictal, and ictal EEG signals, five numbers of rule-based twin SVMs are used. They are TSVM [14], KNN-TSVM [15], KNN-V-RTSVM [16], KNN-STSVM [17], and KWMTSVM [18]. There are many choices of dividing the recorded data into training and test [19] sets. In this study, cross-validation is considered as $k = 10$. For statistical parameter evaluation, k-fold technique is used by using training and testing datasets. The average accuracy and other statistical parameters are experimentally computed after the process repeated for k-times.

2.4 Performance Matrix

The paper evaluates the performance of the proposed methods using parameters [20] sensitivity (S_e), specificity (S_p), accuracy (A_c), positive predictive (PPV), Matthews correlation coefficient (MCC), area under curve (AUC), and execution time. Equations (4)–(8) show parameters taken for validation of the proposed method.

$$S_e = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}\% \tag{4}$$

$$S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}\% \tag{5}$$

$$A_c = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{6}$$

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{7}$$

$$\mathrm{MCC} = \frac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FN} \times \mathrm{FP})}{T1 \times T2} \tag{8}$$

where, $T1 = \sqrt{(\mathrm{TP} + \mathrm{FN})(\mathrm{TP} + \mathrm{FP})}$

$$T2 = \sqrt{(\mathrm{TN} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})}$$

where TP = True positive, FN = False negative,
TN = True negative, FP = False positive

3 Results and Discussion

The block diagram of the proposed method is shown in Fig. 1. For the performance evaluation, we used seven statistical parameters. In this work, $c_1 = 256$, $c_2 = 256$, $c_1 = 64$, $c_2 = 32$ are taken for KNN-STSVM and KWMTSVM classifiers. The p = 8 and p = 0.125 are considered for TSVM and KNN-TSVM. Table 1 shows the statistical evaluation of TSVM classifier. The AUC value is 1 for AE, BE, BD, BCD datasets. The highest specificity is 99.9% for BD-EEG dataset. Table 1 shows the performance

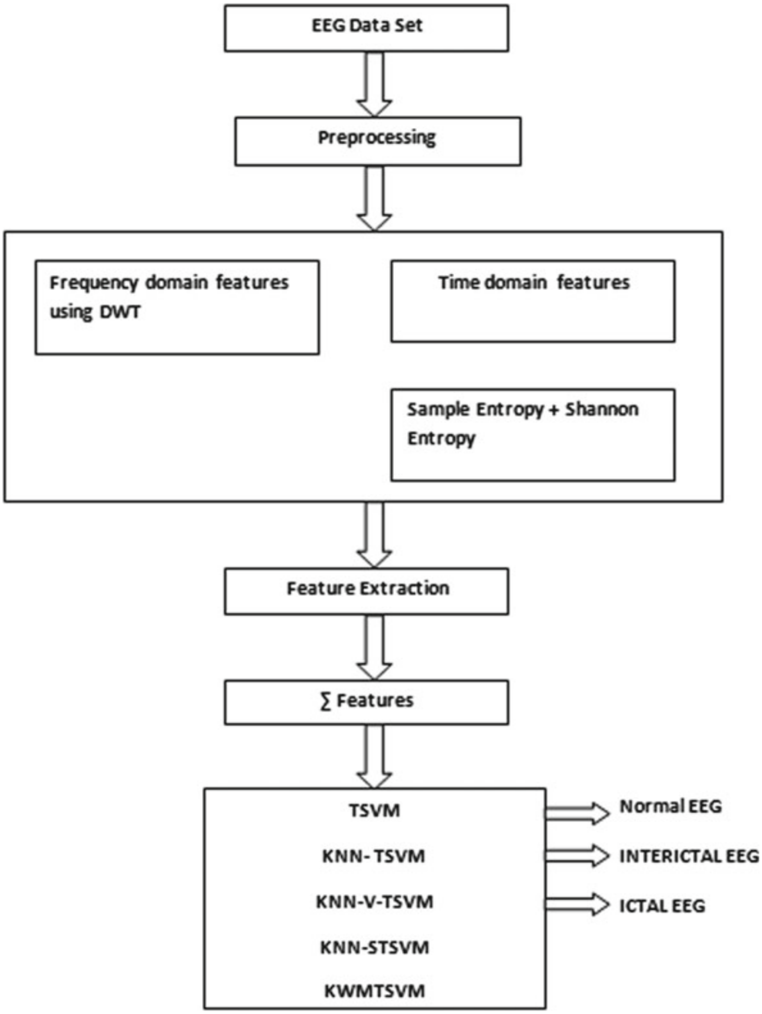


Fig. 1 Block diagram of the proposed model

Table 1 Performance evaluation of the proposed methods

Classifiers	Statistical parameters						
	Sp (%)	Se (%)	Ac (%)	PPV (%)	MCC (%)	Execution time (s)	AUC
TSVM	98.1	91.9	97.9	98.5	91.2	24.62	0.990
KNN-TSVM	91.13	94.83	94.7	91.36	75.94	18.85	0.956
KNN-V-TSVM	99.53	96.72	99.1	99.54	96.02	15.3	0.999
KNN-STSVM	99.3	95.8	98.9	96.1	95.1	19.63	0.996
KWMTSVM	99.5	96.96	99.26	99.57	96.34	7.4	0.999

of KNN-TSVM classifier. In this experimental case, 66.7% of specificity is evaluated for BE set and 97.4% for ABC set. The sensitivity, specificity, accuracy, PPV, and MCC for ABCDE EEG set are 94%, 86.8%, 95.7%, 97%, and 87.7%, respectively. The KNN-v-TSVM classifier performance is evaluated in Table 1. In this case, AUC = 1 for AE, BE, AD, BD, ACE, BCD, and ABC EEG sets. Table 1 shows the overall estimation of KNN-STSVM classifier in which the average specificity is 99.3%, sensitivity is 95.8%, accuracy is 98.9%, PPV is 96.1%, MCC is 95.1%, execution time is 19.63 s, and AUC = 0.996.

Table 1 shows the outperformance of KWMTSVM classifier for every set of experimental works. The specificity, sensitivity, accuracy, PPV, MCC, execution time, and AUC are 99.5%, 96.96%, 99.26%, 99.57%, 96.34%, 7.4 s, and 0.999, respectively. Figure 2 shows the performance analysis of different classifiers. In Fig. 2, it shows that KWMTSVM outperforms in all directions of statistical parameters. Figure 3 shows the scatter plot of KWMTSVM classifier which shows the easiest detection of three types of EEG signals.

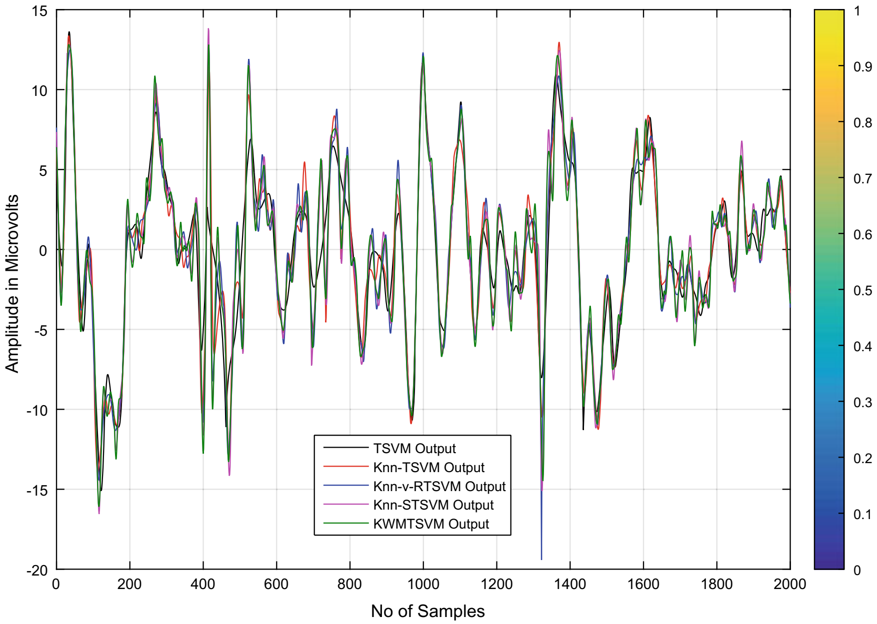


Fig. 2 Plot shows the outputs of different classifiers using linear scale

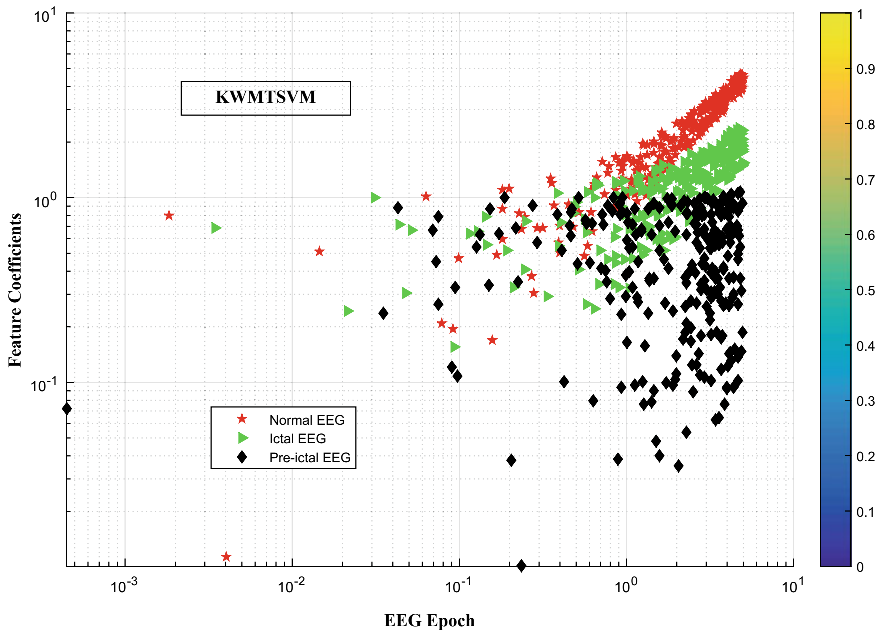


Fig. 3 Scatter plot shows the classification of normal, ictal, and pre-ictal EEG signals using KWMTSVM classifier

4 Comparative Analysis

In this section, the proposed method is compared with the state-of-the-art methods as shown in Table 2. The wavelet features and entropy neural networks are used in [8], having 94% of accuracy. The features like time domain and frequency domains [15] are used to get 99.6% of accuracy. The proposed method DWT-KWMTSVM is with the accuracy 99.26% and with 7.4 s of execution time.

Table 2 Comparative analysis in terms of accuracy (ACC%) of existing methods with proposed methods studied on used database in our work

Reference	Existing method	Accuracy (%)
[8]	Wavelet features	94.00
[9]	Dual-tree, SVM	96.28
[10]	DWT, ANN	96.67
Proposed method	DWT and KWMTSVM	99.26

5 Conclusion

In this research work, three types of features such as time domain, frequency domain, and entropy are extracted. Then, all the features are combined together. Finally, all the features are inputted to different rule-based twin SVMs. The p-value is taken as $p < 0.05$ for all the experimented works. It was observed that KWMTSVM with ABCDE set of data outperforms in all directions. The outcomes in terms of accuracy, sensitivity, specificity, PPV, and MCC are 99.26%, 99.5%, 96.96%, 99.57%, and 96.34%, respectively, with less execution time.

References

1. Niedermeyer, E, Silva, F. L.: Electroencephalography: Basic Principles, Clinical Applications, and Related Fields, 5th edn. Lippincott Williams and Wilkins, London (2005)
2. Sanei, S., Chambers, J.A.: EEG Signal Processing. Wiley, New York (2007)
3. WHO Report, http://www.who.int/mediacentre/factsheets/fs999/en/index.html. Accessed Feb 2011
4. Ruiz, R.A.S., Ranta, R., Louis-Dorr, V.: EEG montage analysis in the blind source separation framework. Bio Signal Process. Control. **6**(1), 77–84 (2010)
5. Coyle, D., McGinnity, T.M., Prasad, G.: Improving the separability of multiple EEG features for a BCI by Neural-time-series-prediction-preprocessing. Bio Signal Process. Control. **5**(3), 196–204 (2010)
6. Ince, N.F., Goksu, F., Tewfik, A.H., Arica, S.: Adapting subject specific motor imagery EEG patterns in space–time–frequency for a brain computer interface. Bio Signal Process. Control. **4**(3), 236–246 (2009)
7. Lehnertz, K.: Non-linear time series analysis of intracranial EEG recordings in patients with Epilepsy–an overview. Int. J. Psychophysiol. **34**(1), 45–52 (1999)
8. Kumar, Y., Dewal, M., Anand, R.: Epileptic seizures detection in EEG using DWT-based ApEn and artificial neural network. Signal, Image Video Process. **8**, 1323–1334 (2014)
9. Das, A.B., Bhuiyan, M.I.H., Alam, S.S.: Classification of EEG signals using normal inverse Gaussian parameters in the dual-tree complex wavelet transform domain for seizure detection. SIViP **10**, 259–266 (2016)
10. Orhan, U., Hekim, M., Ozer, M.: EEG signals classification using the K-means clustering and a multilayer perceptron neural network model. Expert Syst. Appl. **38**(10), 13475–13481 (2011)
11. Andrzejak, R.G., Lehnertz, K., Mormann, F., Rieke, C., David, P., Elger, C.E.: Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. Phys. Rev. E **64**(6), 061907 (2001)
12. Aarabi, A., Wallois, F., Grebe, R.: Automated neonatal seizure detection: A multistage classification system through feature selection based on relevance and redundancy analysis. Clin. Neurophysiol. **117**(2), 328–340 (2006)
13. Farooq, O., Khan, Y. U. et al.: Automatic seizure detection using higher order moments. In: 2010 International Conference on Recent Trends in Information, Telecommunication and Computing. IEEE, pp. 159–163 (2010)
14. Jayadeva, R., Khemchandani, S.: Chandra, twin support vector machines for pattern classification. IEEE Trans. Pattern Anal. Mach. Intell. **29**(5), 905–910 (2007)
15. Ye, Q., Zhao, C., Gao, S., Zheng, H.: Weighted twin support vector machines with local information and its application. Neural Netw. **35**, 31–39 (2012)
16. Xu, Y., Yu, J., Zhang, Y.: KNN-based weighted rough γ-twin support vector machine. Knowl. Based Syst. **71**, 303–313 (2014)

17. Pan, X., Luo, Y., Xu, Y.: K-nearest neighbor based structural twin support vector machine. Knowl. Based Syst. **88**, 34–44 (2015)
18. Yitian, X.: K-nearest neighbor-based weighted multi-class twin support vector machine. Neurocomputing. http://dx.doi.org/10.1016/j.neucom.2016.04.024
19. Chaovalitwongse, W.A., Fan, Y.J., Sachdeo, R.C.: On the time series k-nearest neighbor classification of abnormal brain activity. IEEE Trans. Syst. Man Cybern. Part A Syst. Humans **37**(6), 1005–1016 (2007)
20. Azhar, A.T., El-Said, S.A.: Performance analysis of support vector machine classifiers in breast cancer mammography recognition. Neural Comput. Appl. **24**(5), 1163–1177 (2014)

Artificial Intelligence for Smart Healthcare Management: Brief Study

Subhashree Mohapatra and Tripti Swarnkar

Abstract In recent time, entrepreneurs have started offering smart solutions to monitor healthcare system by using artificial intelligence (AI). It provides a smart alternative to manage the whole healthcare system and helps in accurate prediction and diagnosis of various critical health conditions. Therefore, the advancement and application of AI will change the future healthcare scenario. Even though the AI-based technologies in medical field are advancing rapidly, but in real time, its implementation is yet to be achieved. AI is being used not only in detection of disease but also in allocating professionals, giving home care advises and prescribing medicines. In this article, the importance of AI in the field of healthcare management is shown. Moreover, discussion of several works published in different areas of medicine based on AI during the past few years is carried out briefly. Extending the article, an analysis of the use of AI in health system along with various challenges toward it is done.

Keywords Artificial intelligence · Health care · Management · Diagnosis · Detection

1 Introduction

Health care is a term that defines taking care of human system which not only includes physical care but also mental care [1]. It is a complete cycle to nurture the body through the prevention, diagnosis and treatment of diseases. Any kind of illness or injury as well as any other physical and mental impairment also comes

S. Mohapatra (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), 751030 Bhubaneswar, India
e-mail: subhashreemohapatra68@gmail.com

T. Swarnkar
Department of Computer Application, Siksha 'O' Anusandhan (Deemed to Be University), 751030 Bhubaneswar, India
e-mail: triptiswarnakar@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_37

under health care. Providing healthcare services signifies "the timely use of personal health services to achieve the best possible health outcomes."

Healthcare system comprises a group of people, institutions and machinery resources that give healthcare services to fulfill necessities of the large population [2]. Complete health care is the work of a team in which doctors play an important role in detecting and diagnosis of diseases. As the human body is a complex structure, it is not possible for one person to be expertise of the whole thing. So, to simplify the study, medical field has been divided into various specialties. Some of the broad medical specialty includes immunology, anesthesiology, dermatology, radiology, medicine, internal medicine and many more [3].

AI can be stated as the creation of intelligent behavior in systems in a manner that we would consider it to be smart or human like. In recent years, there has been major contribution of artificial intelligence (AI) in the field of health care [4]. Machine learning is a subset of artificial intelligence technique which has also contributed toward the improvement of health care. Linear and logistic regression that is applied in healthcare research generates a regression function which relates inputs to outputs depending on given data set [5].

Deep learning is another set of techniques that comes under machine learning which is modeled like the nervous system of animals. Deep learning is gaining popularity due to its ability to handle huge amount of complex data [6]. Figure 1 shows the increase in clinical data from year 2013 to year 2020 [7]. As seen from figure, the amount of clinical data is raising in a rapid rate and deep learning is best suited to handle so much of data. As compared to machine learning, deep learning is more valuable in healthcare research as medical data mostly comprises complex data that includes images, videos and unstructured text which are difficult to be processed into explicit features [5].

Smart healthcare system is an emerging trend in the medical field nowadays. Starting from the registration of a patient in a clinic till prescription of medicine, AI has the potential to be introduced in each step of the process. Figure 2 shows the individual step in which a patient is taken care of and the way in which AI is helping

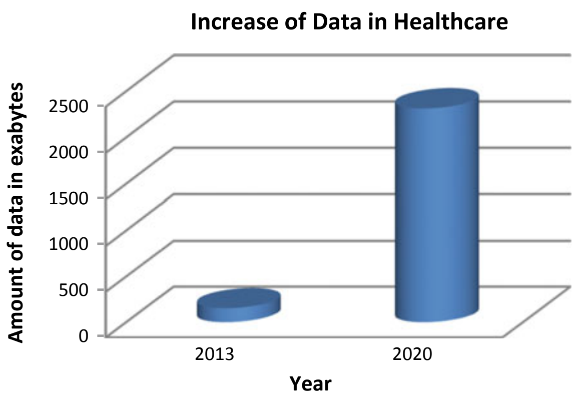


Fig. 1 Plot showing the drastic rise in the healthcare data [7]

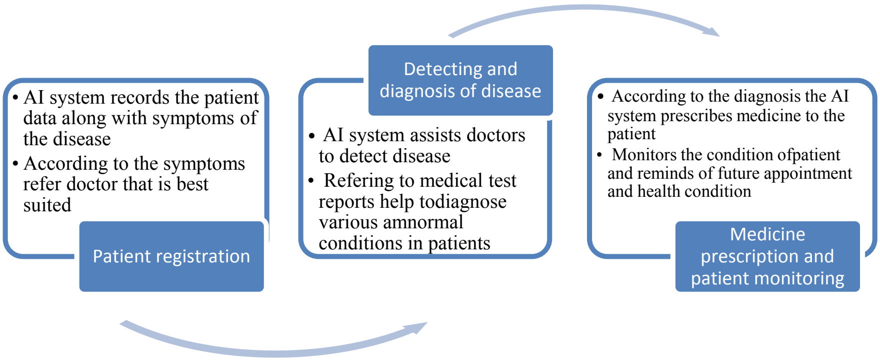
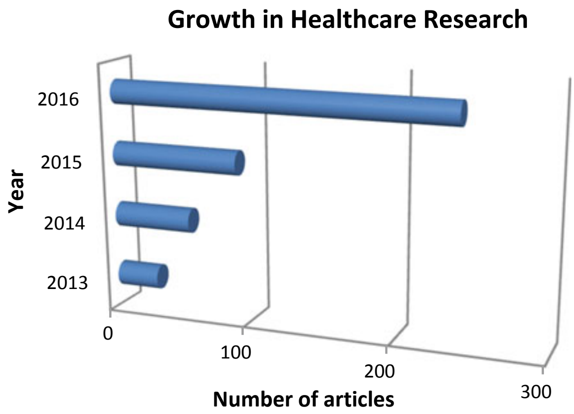


Fig. 2 General steps to make smart healthcare management

to do so. Researchers are working on several areas in which AI could be brought into clinical use. Technologies used in AI are capable to carry out wide range of functions such as detection and diagnosis of different health conditions, predicting future health risk, minimizing medical error and improving efficiency. Bringing AI and deep learning into medicine may not replace healthcare providers but would definitely increase consistency, speed and efficiency. Figure 3 shows the increase of research activity in developing deep learning models in health care [6].

The remaining part of the paper is categorized as mentioned. In the following section, different works done related to AI in health care are discussed. In Sect. 3, a critical analysis regarding the advancement of AI in medical field is done. Finally, in Sect. 4, conclusion of the paper along with future work is presented.

Fig. 3 Plot showing the increase of deep learning application in health care [6]



2 Application of AI in Health care: Study from the Past to Future

From the last two decades, the use of AI in medical field is increasing drastically. AI is being trial and tested in various areas of health care like planning and allocation of resource, detection and diagnosis of diseases, management of health conditions, prescribing drug as well as remote assistance. Figure 4 shows different areas of health care in which AI is used. In this section, a detail discussion is done relating to various areas of research in health care.

A. Healthcare Administration:

As the number of patients is increasing day by day, the burden of maintaining their health records is a challenging task for the medical administrators. AI has been used to execute repetitive and routine tasks of health record entry and review of clinical reports and body scan results as well as allocating resources [8]. This has helped the clinicians to be free to take care of patients. According to an IBM article in 2016 [9], IBM Watson Health and Harrow Council joined hands to use cognitive technology so as to provide personalized health care. It checks for the needs of the individual along with their budget and suggests a care provider. AI is also targeting in improving customer experience. In 2016, Alder Hey Children's Hospital in Liverpool working with IBM Watson has developed a "cognitive hospital" [10]. They have an application to assist the person requirement. The application's aim is to check patient concern before paying a visit, give necessary details on demand and provide healthcare providers with information so as to make the treatment process easy.

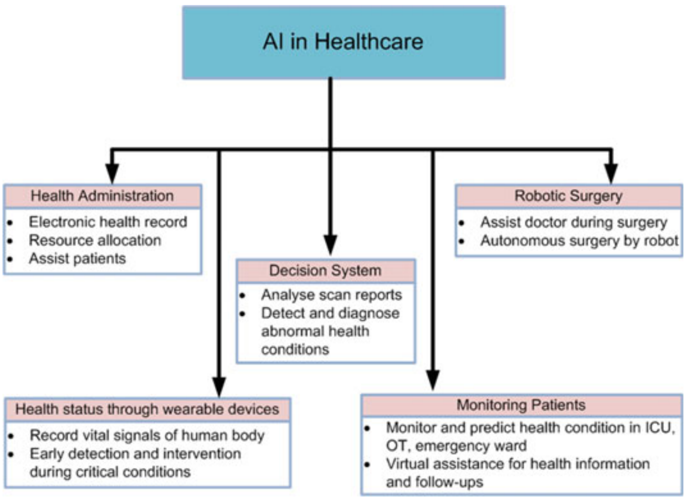


Fig. 4 AI application in different areas of health care

B. Medical Decision System:

Medical decision systems can help to reduce medical errors and increase healthcare efficiency. There is an increase in effort to have practical clinical decision support system [8]. AI could minimize the time as well as cost related to analyzing scan reports, and even more scans can be fetched to give a better treatment. Esteva et al. [11] trained a convolutional neural network on 129,450 medical images of skin which attained an accuracy level of dermatologist in diagnosing skin malignancy. In many areas such as the detection of lung nodules using CT scan images, the diagnosis of pulmonary tuberculosis and common lung diseases with chest radiography and finding abnormal growth in breast using mammography images, application of AI has achieved a professional-level diagnostic accuracy [3]. Gulshan et al. trained convolutional neural network models to identify referable diabetic retinopathy and diabetic macular edema using 128,175 retinal images [12]. Misawa et al. designed a deep learning model that was trained on 306 polyp-negative and 105 polyp-positive videos. The model was also tested on different data sets which was capable of finding 94% of polyps with a false positive detection rate of 60% [13].

C. Monitoring of Patient:

Some of the crucial places like cardiac wards, ICUs, emergency wards and operation theaters need continuous monitoring of patients' health conditions. Huge amount of data is generated in these sections which could be efficiently used to develop AI models so as to have alert generators in case of critical conditions [3]. Churpek et al. [14] developed a model for predicting cardiac attack using vital health signs. Lundberg et al. [15] proposed a system that could be helpful to anesthesiologists for interpretation of hypoxemia condition during operations. Information devices or chatbots developed using AI are being implemented for managing some health conditions. In 2017, IBM and Arthritis Research UK announced that patients with arthritis could be able to interact with Arthritis Virtual Assistant for any advice relating to their medicine, diet and exercise [16].

D. Robotic Surgery:

Nowadays, various industries and biomedical laboratories are using robotic systems very often. But in case of medical organizations, adopting and implementation of robotic system are considerably less [3]. Many healthcare providers do prefer robot-assisted surgery where the systems are controlled by the practitioner for performing the operation. Gomes in [17] mentioned in his article that Food and Drug Administration approved da Vinci Surgical System; the surgeon handles the robot through a console for a minimally invasive operation. More research is going on for developing automated methods to add-on to the surgical operations.

E. Inferring Health Status Through Wearable Devices:

In the present day, there are various wearable devices available which can record medical signals like pulse rate, heartbeat, tremor in voice, limb moments of a person [3]. These signs are very useful in early detection and warning of different health conditions. Li et al. developed a device used for predicting and tracking infection by recording skin temperature and rate of heartbeat [18]. Pastorino et al. [19] described how wearable devices are useful in detecting Parkinson's disease by symptoms like unusual hand movement, speech pattern or tremor.

3 Challenges: Critical Analysis for Future

As the AI techniques are rapidly progressing into healthcare services, at the same time, many challenges are still to be handled [8]. Till now, most of the medical research done using AI has been carried out on retrospective data collected from different healthcare centers. But in reality, prospective medical trials are required to test the efficiency of the clinician AI models. Prospective studies will obviously help to recognize the weakness or lacuna of the AI models in the real-world situations [20]. To bring AI into healthcare services, there are various challenges that need to be studied, analyzed and have proper solutions to them. In this section, multiple issues are discussed.

A. Social, economic and legal challenges:

Numerous economic, social and legal issues would come up when AI systems take positions in regular clinical practices. AI systems can only have its full potential when it would be integrated into day-to-day use in the healthcare centers [21]. Before deployment of AI systems into health organizations, the system has to be properly validated and certified. So, a clear regulation and guideline are necessary for the certification of the AI systems [20]. The team which designs, develops and maintains the system needs to follow the guidelines strictly.

Implementation of AI systems in medicine would require a hefty amount of money. Government organizations as well as private firms have to take necessary steps by providing enough funding so as to initiate the AI application in health care. Another major concern is the legal challenges. In case of clinical negligence related to automated decision systems, malpractice related to AI application, there have to be clear legal guidelines about who will be held liable [3]. There should be proper insurance policies when health care will be carried out majorly by AI systems.

B. Technical challenges:

As the AI systems rapidly enter the world of health care, there will also be many technical challenges to be faced. Different clinical environments have different types of external disturbance and noise which can affect the working of AI model. An AI system developed for one center may not work the same way in other hospital [22]. So, the application that will be developed should be such that it would give the same performance in all the environments. The amount of digital data related to patient's health record is huge, so the collection and storage of these sensitive data remain a challenge. Researchers are working on this issue so that the critical information of the patients could be securely handled [23]. It is our responsibility to design and develop security systems and encryption methods to keep the data safe. Another issue is the interpretation of the result given by the AI models. Interpretation of models is more challenging in case of deep neural network than machine learning models [3]. More research is going on as to how to interpret deep learning models trained on data other than images.

C. Ethical challenges:

Understanding the ethical issues is highly necessary while implementing AI tool in medicine. Some of the important questions to be answered are like will the clinicians accept the AI applications? How the end users take the decision given by AI machines? Physician may be reluctant in accepting the AI applications in health care [5]. Even when clinicians agree to work on AI systems, the next thing will be how to train and make them understand about the working of the AI models. Training of the healthcare providers is highly required so that they will clearly understand the system. Most efficient and easily understandable method has to be developed to train the physicians. A good way to do this is involving the clinicians during the development and testing of models. By doing so, the physicians would have a better understanding as well as a feel of trust on the AI applications in various fields of health care [8]. PricewaterhouseCoopers (PwC) did a survey around twelve countries in 2016 to know the percentage of population willing and not willing to have AI application in their healthcare service [24]. Figure 5 shows the result of the survey. From the figure, it is seen that around 38% of the population is unwilling to incorporate AI in health care. This is a big challenge to work on.

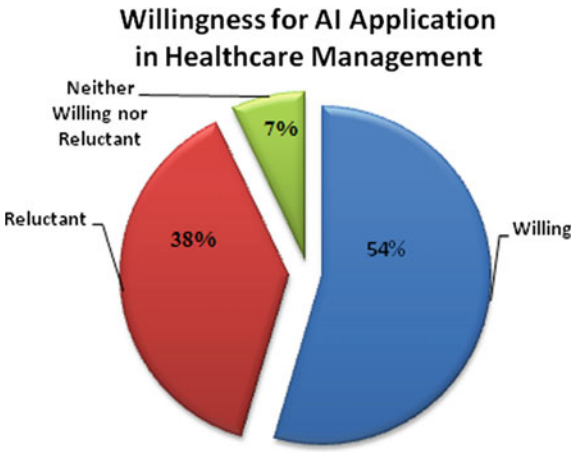


Fig. 5 Survey result done by PwC in 2016 [24]

4 Conclusion and Future Work

Undoubtedly AI system in healthcare management will bring a revolutionary change in the field of medicine. From the past few years, rigorous research is being done to develop different AI models. The developed techniques could be used not only in detection and diagnosis of disease but also in health assistance, drug prescription and robotic surgeries. At present, many start-ups and organizations are working on specific areas of medicine for developing application to solve many problems. The availability of wearable devises and remote health assistance has developed a sense of responsibility in human being regarding their health. Even though AI is changing the face of healthcare system, there are many challenges which have to be taken care of. Researchers, legal systems and healthcare organizations have to come together to find out solutions to these issues. As it is studied that in each step of medical examination of a patient, AI has the scope of implementation, so new techniques could be developed to assist the doctors in the hospitals. Increase in the number of patients in recent times has put more pressure on doctors and experts to manually analyze several reports and diagnose the disease. So, to assist the doctors, AI systems could be developed which can detect and diagnose diseases in patients. In future, authors will be analyzing different areas in which AI systems can be implemented and develop an AI system to solve a specific problem.

References

1. Khan, S. Yairi, T.: A review on the application of deep learning in system health management. Mech. Syst. Signal Process. **107**, 241–265
2. White, F.: Primary health care and public health: Foundations of universal health systems. Med. Princ. Pract. **24**(2), 103–116 (2015)
3. Yu, K. H., Beam, A. L, Kohane, I. S.: Artificial intelligence in healthcare. Nat. Biomed. Eng. **2**(10), 719–731
4. Gambhir, S., Malik, S.K., Kumar, Y.: Role of soft computing approaches in healthcare domain: A mini review. J. Med. Syst. **40**(12), 287 (2016)
5. Alagappan, M., Brown, J.R.G., Mori, Y., Berzin, T.M.: Artificial intelligence in gastrointestinal endoscopy: The future is almost here. World J. Gastrointest. Endosc. **10**(10), 239 (2018)
6. Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., Wang, Y., Dong, Q., Shen, H., Wang, Y.: Artificial intelligence in healthcare: Past, present and future. Stroke Vasc. Neurol. **2**(4), 230–243 (2017)
7. The Future of Healthcare is Tied to AI Big Data, https://cmte.ieee.org/futuredirections/2018/05/18/the-future-of-health-care-is-tied-to-big-data/. Last accessed 2019/10/15
8. Reddy, S., Fox, J., Purohit, M. P.: Artificial intelligence-enabled healthcare delivery. J. R. Soc. Med. **112**(1), 22–28 (2019)
9. IBM And Harrow Council To Bring Watson Care Manager To Individuals In The Uk, https://Newsroom.Ibm.Com/2016-09-07-Ibm-And-Harrow-Council-To-Bring-Watson-Care-Manager-To-Individuals-In-The-Uk. Last accessed 2019/10/01
10. Alder Hey Children's Hospital Set To Become Uk's First 'Cognitive' Hospital, https://Alderhey.Nhs.Uk/Contact-Us/Press-Office/Latest-News/Alder-Hey-Childrens-Hospital-Set-Become-Uks-First-Cognitive-Hospital. Last accessed 2019/10/01

11. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**(7639), 115 (2017)
12. Gulshan, V., Peng, L., Coram, M., Stumpe, M.C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., Kim, R.: Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. Jama **316**(22), 2402–2410 (2016)
13. Misawa, M., Kudo, S. E., Mori, Y., Cho, T., Kataoka, S., Yamauchi, A., Ogawa, Y., Maeda, Y., Takeda, K., Ichimasa, K. And Nakamura, H.: Artificial intelligence-assisted polyp detection for colonoscopy: Initial experience. Gastroenterology, **154**(8), 2027–2029 (2018)
14. Churpek, M. M., Yuen, T. C., Huber, M. T., Park, S. Y., Hall, J. B., Edelson, D. P.: Predicting cardiac arrest on the wards: A nested case-control study. Chest **141**(5), 1170–1176 (2012)
15. Lundberg, S.M., Nair, B., Vavilala, M.S., Horibe, M., Eisses, M.J., Adams, T., Liston, D.E., Low, D.K.W., Newman, S.F., Kim, J., Lee, S.I.: Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. Nat. Biomed. Eng. **2**(10), 749 (2018)
16. Arthritis Research Uk Introduces Ibm Watson-Powered 'Virtual Assistant' To Provide Information And Advice To People With Arthritis. See https://www-03.Ibm.Com/Press/Us/En/Pressrelease/51826.Wss. Last accessed 2019/10/03
17. Gomes, P.: Surgical robotics: Reviewing the past, analysing the present, imagining the future. Robot. Comput. Integr. Manuf. **27**(2), 261–266 (2011)
18. Li, X., Dunn, J., Salins, D., Zhou, G., Zhou, W., Rose, S.M.S.F., Perelman, D., Colbert, E., Runge, R., Rego, S., Sonecha, R.: Digital health: Tracking physiomes and activity using wearable biosensors reveals useful health-related information. PLoS Biol. **15**(1), E2001402 (2017)
19. Pastorino, M., Arredondo, M. T., Cancela, J., Guillen, S.: Wearable sensor network for health monitoring: The case of parkinson disease. In journal of physics: Conference series, 450(1), 012055 (2013). Iop Publishing
20. Yu, K. H., Kohane, I. S. Framing the challenges of artificial intelligence in medicine. Bmj Qual. Saf. **28**(3): 238–241 (2019)
21. Longhurst, C. A., Parast, L., Sandborg, C. I., Widen, E., Sullivan, J., Hahn, J. S., Dawes, C. G., Sharek, P. J.: Decrease In hospital-wide mortality rate after implementation of a commercially sold computerized physician order entry system. Pediatrics **126**(1), 14–21 (2010)
22. Obermeyer, Z., Emanuel, E.J.: Predicting the future—big data, machine learning, and clinical medicine. N. Engl. J. Med. **375**(13), 1216 (2016)
23. Lee, L. M., Gostin, L. O.: Ethical collection, storage, and use of public health data: A proposal for a national privacy protection. Jama **302**(1), 82–84 (2009)
24. Enabling Better Healthcare With Artificial Intelligence, https://Usblogs.Pwc.Com/Emerging-Technology/Ai-In-Healthcare/. Last accessed 2019/10/08

Leaves Shape Categorization Using Convolution Neural Network Model

Sushruta Mishra, Debjit Koner, Lambodar Jena, and Piyush Ranjan

Abstract In our natural ecosystem, there are thousands of varieties of trees, and to discriminate among them is very tough. People who study plants can able to recognize the kind of tree at a glance by using the features of the leaf. Leaf types can automatically be classified using machine learning. Studied extensively in 2012, based on deep learning this is a swiftly growing field. Large amounts of data are used as deep learning is a more practical technique. Nowadays based on its developments in hardware and big data, deep learning is all about self-learning techniques. In this research, a convolution neural network-based image processing model is proposed for the effective classification of leaves. A total of 2500 samples of leaves were gathered and partitioned into training and testing set in varying ratios. The result shows that our proposed CNN-based leaves classification model shows optimal output with respect to accurately predicting the category of the leaf.

Keywords Convolution neural network · Leaves classification · Deep learning · Image augmentation · Accuracy rate

S. Mishra · D. Koner · P. Ranjan
School of Computer Engineering, KIIT University, Bhubaneswar, India
e-mail: mishra.sushruta@gmail.com

D. Koner
e-mail: debjitkoner@gmail.com

P. Ranjan
e-mail: piyushupadhyaya22@gmail.com

L. Jena (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be) University, Bhubaneswar, India
e-mail: lambodarjena@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_38

1 Introduction

We may see many rare kinds of trees when leaving a city and entering the outskirts. The majority of us don't know most of the trees or plants which grow outskirts of the city, trees or plants that grow on the city street can be identified by us. However, there are approximately about 100,000 kinds of different trees and plants on this planet, in which 25% of them are shrubberies. In the topical region, we found many kinds of trees but still, limited botanical research has been performed till now. For this, it's still believed that many rare kinds of trees are not even discovered until now. It is clear that it's a complex process of identifying those rare species of trees. There are very similar types of leaves the same shape, trees of the same shape, and even in young fruit shape. There are similar kinds of flower shapes, and whenever the calyx is attached those trees can only be identified, or petals are relatively inverted. Moreover, at a particular time, we can easily distinguish some particular trees; for example, when they bear fruit or bloom. These kinds of trees can be identified, by using significant information, including the leaf shape, the leaves shape that is directly attached to the branches, shape of the branch, whole tree shape, size of the tree, shape of the flower, time of flowering and fruit. The subject like molecular biology, morphology anatomy, cell biology, or phytochemistry, by using this subject we can differentiate plants without taking much time. However, by using this method it is very impracticable for general people to identify the names of trees or plants, for example, when we walk in a forest.

All these above methods are focused on visualization of plants and their differences in geometric characteristics modeling and the advantages own by them. Be that as it may, a large portion of them streamlined the leaf or vein to a surface or a bend, which is a long way from the real morphology of the leaf and appropriate for rendering complex scenes. They are unsuccessful to show the geometric model of the leaf accurately. The leaf isn't the finished surface, particularly the veins that are a direct chamber with an adjustment of thickness. It is important to devise another strategy to manipulate the exactness of the geometric displaying of a leaf vein. In light of this thought, information on the leaf edges, horizontal veins, and fundamental veins are procured and utilizing Interpolation NURBS, and Interpolation Ball B-Spline Curve (IBBSC) a surface model is recreated. Hence we have proposed a deep learning-based CNN model to accurately categorize the shape of leaves.

2 Dataset

The task will be based on the PlantView dataset which focuses on 250 herb and tree species from the France area. It contains 2677 pictures belonging each to one of the two following categories:

Geometry shape (or uniform background) (42%): exclusively pictures of leaves in front of a white or colored uniform background produced with a scanner or a camera with a sheet.

Non- Geometry (for most of the time cluttered natural background") (58%): free natural photographs of different views on different sub-parts of a plant into the wild.

3 Literature Survey

Miao et al. proposed a proof hypothesis put together rose arrangement [1] based with respect to numerous highlights of roses. Gu et al. attempted leaf acknowledgment utilizing skeleton division by wavelet change and Gaussian addition [2]. Wang et al. utilized a moving middle focus (MMC) hypersphere classifier [3]. The creators proposed a used dynamic programming algorithm for calculation of the leaf shape coordinating [4]. Ye et al. contrasted the similitude between highlights with order plants [5]. Saitoh et al. consolidated bloom and leaf data to order wildflowers [6]. Heymans et al. proposed the use of ANN to arrange opuntia species [7]. Du et al. presented shape acknowledgment dependent on an outspread premise probabilistic neural system which is prepared by the symmetrical least-square calculation (OLSA) and upgraded by recursive OLSA [8]. Through adjusted Fourier descriptors of leaf shape, plant acknowledgment is performed.

Past work has a few disservices. Some are just pertinent to specific species [1, 7]. As a specialist framework, a few strategies look at the closeness between highlights [5, 8]. It requires the pre-handling work of people to enter the keys physically.

Utilizing the bit descriptor (KDES) as another element extraction procedure, Le et al. (2014) a completely robotized plant ID framework being created [9].

Arai et al. (2013) accomplished a precision of 95.8% utilizing support vector machines (SVM) classifier [10]. The size of each image was 256*256 pixels. Du et al. (2013) acknowledgment and characterization plant leave dependent on fractal measurement highlights dependent on leaf shape and vein designs for the [11]. Create a texture signature for a leaf and the Linear Discriminant Analysis (LDA) algorithm utilizing a volumetric fractal measurement, Backes et al. (2009) beat the customary methodologies that depended on Gabor channels and Fourier examination [12].

4 Proposed Work

Using feature extraction and selection a lot of work time can be saved by using deep learning. By equating the outdated method with the new method, we have proposed a new model to detect leaf shape (geometry or non-geometry) images through a recognition system with a CNN model, which incorporates two principal forms which include image pre-processing and learning (see Fig. 1).

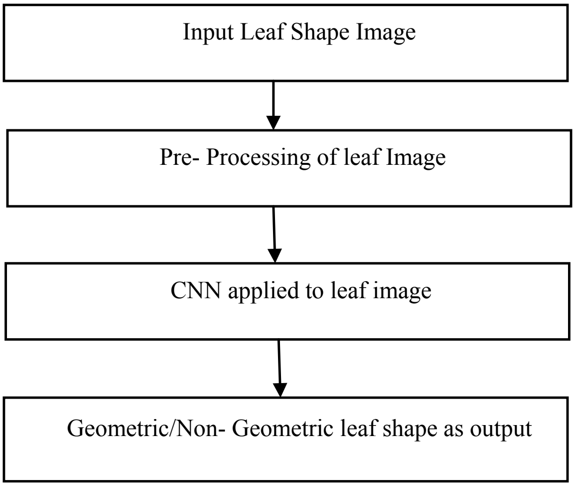


Fig. 1 Proposed CNN model for leaf categorization

Normalization of images holds a big issue when it comes to training a CNN model. Most of the images present in the dataset are not in a standard size which can create problems while training the model. Therefore, pre-processing of images is done where the original size of the image, i.e., 1600×1200 is converted into images of size 64×64 for training and testing purposes. As per the block diagram, the pre-processing block is divided into a test set and training set. The operations involved in the training dataset include the following.

Image augmentation: This technique is used in deep learning models to increase the model's accuracy and performance. Too much data in the training set and test set cause overfitting. To overcome these we use augmentation of images. In this method, the training images are automatically created using many processed which is (i) random rotation (ii) random flip.

Rescaling: It is an operation that moves data from one numerical range to another by simple division using a predefined constant. In deep neural networks, we restrict input to the range from 0 to 1 due to possible overfitting, optimization, stability issues.

Shearing: Shear mapping or shearing displaces each point in the vertical bearing by a sum relative to its good ways from an edge of the picture. However, it is not mandatory that the image should be in vertical it can be arbitrary also.

Zoom: It gives some random zoom in or zoom out. The zoom_range parameter this parameter controls the zooming factor of the image. For instance if zoom_range parameter is equal to 0.2 it means range chosen from [0.2, 1.2].

Flip: It turns over the image horizontally or vertically. We have used a horizontal flip in this model. The image is being flipped into concerning the vertical axis. We can turn on or off the horizontal_flip by using horizontal_flip_parameter.

In the test set, the images which are given in Fig. 2 is utilized here are overwhelmingly put away in an RGB (Red Green Blue) design. Here the image is portrayed into three-dimensional shape (or three-channel) exhibit. One dimension is for channels

Fig. 2 **a** Geometry shape and **b** non-geometry shape

(red, green, and blue colors) and two other dimensions are the spatial dimension. Thus, every pixel is encoded through three numbers. Each number is generated by storing an 8-bit unsigned integer type (0–255). Rescaling was used to handle the stability, optimization, and overfitting issues in the model.

In our study, a convolution neural network which is a deep neural network model is used to predict the leaf shape images. The framework of the CNN model is depicted in Fig. 3.

Convolution Layer: It a function derived from two given functions by integration which expresses how the shape of one is modified by the other. We used 32 feature detectors which helped to extract feature information from the images (stored in a 3 × 3 dimensional shape). Feature detectors are "Kernel" or "Filter". Two convolution layers ensure the number of feature information extractions which is responsible to increase the accuracy and performance of the model.

Input shape: It takes input of the image with a 64 × 64 dimension each with three channels as it is a color image.

Activation function: It is an activation that is responsible to acquaint non-property properties with the system by mapping the inputs the response variables. Rectified Linear Unit transformation function, i.e., ReLU is used here as an activation layer because it helps to make sure that there is no negative pixel value in the feature map.

Max-Pooling: The pooling layer is matrix but it helps to sort the matrix created by feature detector in the convolution layer. It takes important information from the convolution layer and leaves the rest. Here, we have used a window of size 2 × 2 and stride 2 for pooling.

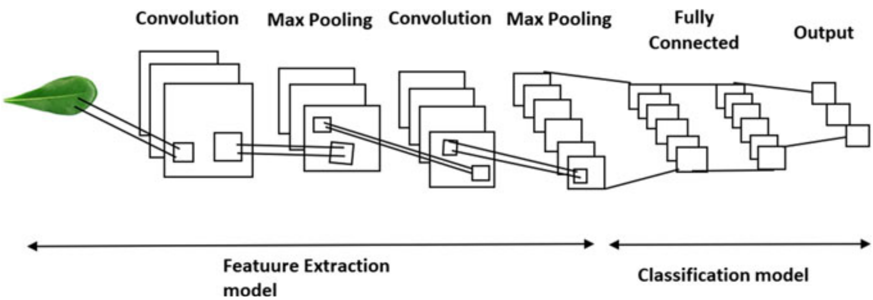


Fig. 3 The architecture of the CNN model

Flattering: It is a utility operation that flattens an input shape to a simple vector output. It is used to flatten the pooled feature map into a column. Here shrink image is taken as input and then converted into a vector for classification.

Full Connection: To classify the classic artificial neural network as an image classifier is a non- linear problem so a hidden layer or fully connected layer is made with 128 nodes. This number is not too large and not too small for the system which helps the model to perform well.

Loss Function: It is an important function of artificial neural networks. It provides non-negative value, where the loss function decrease with increasing the robustness of the model. Here, binary- cross-entropy is used to calculate the loss output.

Optimizer: It helps to update weights in the neural network. Adam optimizer has been used to update network weights.

Compiling the CNN: It is used to stack up all the previously mentioned layers and compile the model into one.

Prediction: Finally, the binary encodings show the prediction of the leaves as to which class a particular leaf belongs, i.e., geometry shape or non-geometry shape

5 Result Analysis

The leaf images are categorized using the convolution neural network model. Here, we divide the dataset into three parts, i.e., 40, 50, 60% testing set, with 25 epoch each. Figure 4 represents the total accuracy obtained in each part. Figures 4 and 5. Figure 6 represents the graph between Value_Loss vs Value_Accuracy, respectively.

Then we divided the test set and training set into three different parts (i.e., 60, 50, 40% into a training set and test set) followed by the same architecture. In Fig. 4, the accuracy rate is shown with respect to the different training and testing set variations. To represent the model differently we took differently test set and training set size

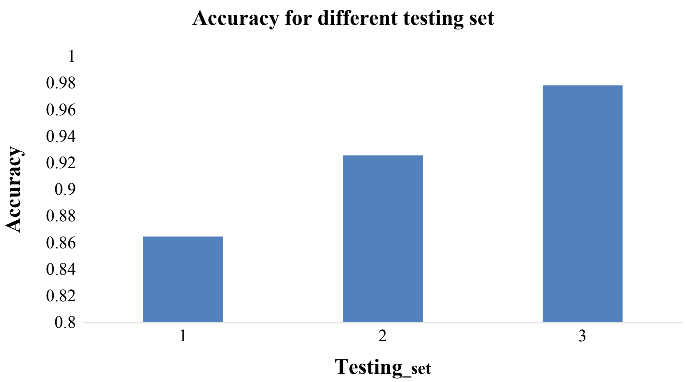


Fig. 4 Results of the proposed CNN model with different training/testing set ratios

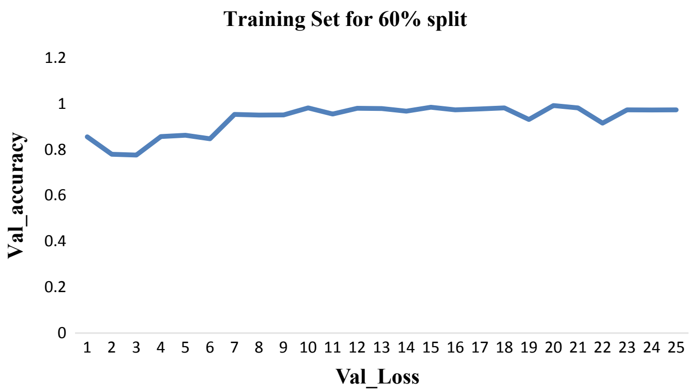


Fig. 5 Graph of value_loss versus value_accuracy for test set 60%

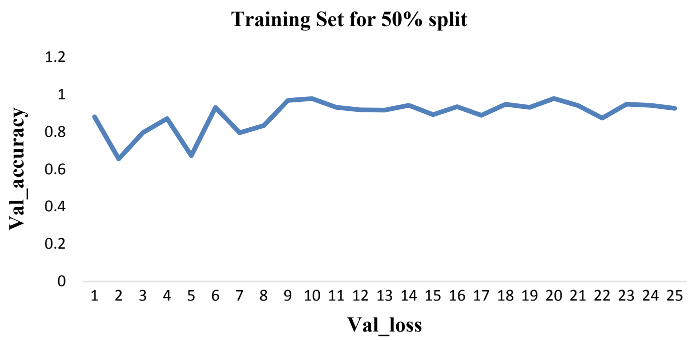


Fig. 6 Graph of value_loss versus value_accuracy for test set 50%

and we slowly increase the test set size to see to compare the model accuracy with more training size.

In Fig. 5, an accuracy of about 97.83% is obtained while in Fig. 6 we get an accuracy rate of about 92.56%. In Fig. 7, we get an accuracy of about 86.47%. These figures are represented by a line graph where we compare the value_loss that find the model error (i.e., help us to understand whether our model is overfitting, under-fit or it's a good model), value _accuracy help us to test the model accuracy and the blue line in the graph below represents the value_loss. All the graphs given below show different classification accuracy rates to train the model into 3 times with different partitions of the same dataset.

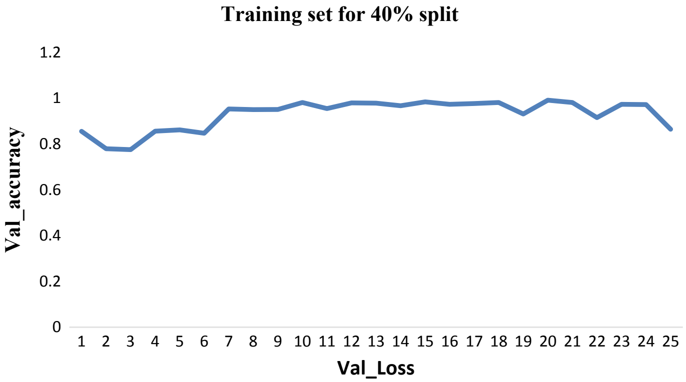


Fig. 7 Graph of value_loss versus value_accuracy for test set 40%

6 Conclusion

In this paper, we proposed a CNN-based image processing method to categorize leaves shape (geometry or non-geometry) using the deep learning model and created three different models each by changing the data sample size of training and testing sets. We assessed the working of each model as per the discoloration or damage to leaves. The accuracy rate accomplished was more noteworthy around 94%, while 15% of the leaf was harmed. In future research, we will endeavor to perceive leaves connected to branches, to build up a visual framework that can imitate the technique utilized by people to distinguish plant types. This classification will help us to know that from where the leaves belong (a hot area or cold area) which will help to let us know which leaf can survive more in a cold area and which can survive in a hot area.

References

1. Miao, Z., Ganelin, M. H., Yuan, B.: An over-based rose variety recognition system. Eng. Appl. Artif. Intell. **19** (2006)
2. Gu, X., Du, J. X., Wang, X. F.: Leaf recognition based on the combination of wavelet transform and Gaussian interpolation. In: Proceedings of International Conference on Intelligent Computing, ser. LNCS 3644. Springer (2005)
3. Wang, X. F., Du, J. X., Zhang, G. J.: Recognition of leaf images based on shape features using a hypersphere classifier. In: Proceedings of International Conference on Intelligent Computing 2005, ser. LNCS 3644. Springer (2005)
4. Du, J. X., Huang, D. S., Wang, X. F., Gu, X.: Computer-aided plant species identification (caps) based on leaf shape matching technique. Trans. Inst. Meas. Control. **28** (2006)
5. Ye, Y., Chen, C., Li, C. T., Fu, H., Chi, Z.: A computerized plant species recognition system. In: Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, Hong Kong (2004)
6. Takeshi Saitoh, T. K.: Automatic recognition of wildflowers. In: Proceedings of 15th International Conference on Pattern Recognition (ICPR'00), vol. 2 (2000)

7. Heymans, B. C., Onema, J. P., Kuti, J. O.: A neural network for opuntia leaf-form recognition. In: Proceedings of IEEE International Joint Conference on Neural Networks (1991)
8. Du, J., Huang, D., Wang, X., Gu, X.: Shape recognition based on radial basis probabilistic neural network and application to plant species identification. In: Proceedings of 2005 International Symposium of Neural Networks, ser. LNCS 3497. Springer (2005)
9. Le, T. L., Tran, D. T., Hoang, V. N.:. Fully automatic leaf-based plant identification, application for Vietnamese medicinal plant search. In: Fifth Symposium on Information and Communication Technology, Hanoi, Vietnam, pp. 146–154 (2014)
10. Arai, K., Abdullah, I.N., Okumura, H.: Identification of ornamental plant functioned as medicinal plant-based on redundant discrete wavelet transformation. Int. J. Adv. Res. Artif. Intell. **2**(3), 60–64 (2013)
11. Du, J.X., Zhai, C.M., Wang, Q.P.: Recognition of plant leaf image based on fractal dimension features. Neurocomputing **116**, 150–156 (2013)
12. Backes, A.R., Casanova, D., Bruno, O.M.: Plant leaf identification based on volumetric fractal dimension. Int. J. Pattern Recognit. Artif. Intell. **23**(6), 145–1160 (2009)

Thyroid Disorder Analysis Using Random Forest Classifier

Sushruta Mishra, Yeshihareg Tadesse, Anuttam Dash, Lambodar Jena, and Piyush Ranjan

Abstract Nowadays, diseases are increasing due to the lifestyle of human beings. Thyroid disorder is also increasing. There are two types of thyroid disorder. Hyperthyroidism occurs due to the lack of thyroid hormone in the blood, and hypothyroidism occurs due to the availability of the excess amount of thyroid hormone in the blood. In order to diagnose the disorder, it needs enough experience and knowledge. This makes the diagnosis of thyroid disorder difficult. Data mining technique simplifies the problem with better accuracy. In this proposed work to analyze the problem of hypothyroidism, random forest classification algorithm is used and its performance is compared with other algorithms such as sequential minimal optimization (SMO), decision table, and *K*-star classifier. The result shows that random forest and decision rule have good performance for the prediction of thyroid disorder.

Keywords Thyroid disorder · Random forest · Classification · Machine learning · Accuracy rate

S. Mishra · Y. Tadesse · A. Dash · P. Ranjan
School of Computer Engineering, KIIT University, Bhubaneswar, India
e-mail: mishra.sushruta@gmail.com

Y. Tadesse
e-mail: tadehareg@gmail.com

A. Dash
e-mail: anuttam.dash@gmail.com

P. Ranjan
e-mail: piyushupadhyaya22@gmail.com

L. Jena (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be) University, Bhubaneswar, India
e-mail: lambodarjena@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_39

1 Introduction

Thyroid disorder is becoming a worldwide problem especially for women than men. The main cause of this problem is the lifestyle of human beings. Thyroid hormone is secreted from the thyroid gland, and it has many functions such as metabolism rate control, burn calories, and protein. It also regulates the excessive secretion of hormone from other hormonal glands. There are two types of thyroid disorder. The first one is hyperthyroidism which occurs due to the availability of excess thyroid hormone in the blood, and the second one hypothyroidism is due to low thyroid hormone in the blood. Both hyperthyroidism and hypothyroidism have their own negative side effects to the normal functioning of the human body. In medical science field, diagnosis of health disorder at the early stage is the most challenging task for most healthcare experts. Especially, in case of thyroid disorder, it needs an experienced and knowledgeable person. It also may have similar symptoms with other health problems. Unless thyroid disorder is treated at an early stage, it may lead to other health problems and sometimes death. Data mining techniques contribute to a lot in medical data analysis for different diseases. In this paper, also, we use data mining technique for hypothyroid disorder analysis. Classification and diagnosis of hypothyroid disorder using random forest algorithm and comparison of its performance with other algorithms form the basis in this study. From the selected hypothyroid data set, we use data classification using random forest algorithm on WEKA tool and performance of the classifier is tested using K-fold cross-validation; thereby, its performance is compared with sequential minimal optimization, decision table, and K-star algorithms. Finally, the overall result is discussed.

2 Literature Survey

Various research works have been conducted in this aspect of prediction of thyroid disease using several data mining algorithms in recent times. Numerous machine learning-based techniques have been used in the previous research works with the aim to achieve better accuracy rate and preciseness of predicting thyroid disorder. In [1], Polat and his colleagues developed an artificial immune framework for proper diagnosis of thyroid disease, thereby achieving 81% accuracy rate. Keles in [2] developed a neuro-fuzzy-based expert model for classification of thyroid disease and obtained a prediction accuracy of 95.33% with efficient implementation. In [3], Temurtas performed successfully diagnosis of thyroid disorder by combining Levenberg Marquardt-LM classifier with the backpropagation algorithm and determined the accuracy rate of 93.19%. A support vector machine-based model was combined with generalized discriminant analysis (GDA) techniques in [4] for analyzing thyroid disorder risks, thereby getting an optimal accuracy of 91.86%. An expert system model referred to as Fisher Score Particle Swarm Optimization Support Vector Machines (FS-PSO-SVM) was proposed by ChenHui-Ling in [5] and was applied on thyroid disease data set gathering a good output with acceptable accuracy rate. In [6], a comparative analysis between SVM and KNN classifiers was done by Nikita Singh and her colleagues, and it was concluded that SVM was better than KNN in terms of accuracy rate of 84.62%. Two parallel variations of source code for texture-oriented segmentation have been developed by Edgar Gabriel et al. [7] for thyroid FNAC images, thus achieving a fully automated model for thyroid prediction. In [8], an automated expert-based segmentation technique was suggested by Preeti Aggawal et al where all results of various algorithms were summarized for both lung disease and thyroid risk disorders. In [9], Sushruta Mishra et al. developed and implemented a new machine learning technique for dengue prediction which resulted in a good accuracy rate with minimum latency. In paper [10], backpropagation algorithm is proposed for checking the presence of thyroid. By using backpropagation of error, artificial neural network is developed to detect thyroid.

3 Proposed Work

Hypothyroid disorder analysis is done by selecting the data set from the UCI repository. The data set is collected from the UCI repository, and it has 3772 instances. The negative category consists of 3481 instances, the compensated hypothyroid 194, primary hypothyroid 95 instances, and secondary hypothyroid category has 2 instances. Its attributes are 30 from which the last attribute will be used for data classification. The data set has 0 percent missing value. To retrieve the information from the data, it passed through data preprocessing. Data preprocessing includes removal of stop words, stemming, token generation, and frequency matrix formation. After preprocessing of the data, it is converted to ARFF format and classified with some vital classifiers which include random forest, SMO, *K*-star, and decision table.

Random forest classifier consists of a number of decision trees that coordinate to predict a class. Each individual tree splits out a class prediction, and the most votes' class becomes a model prediction. Sequential minimal optimization (SMO) is support vector machine; its quadratic programming problem is solved. It is an improved support vector machine, and its quadratic problem is partitioned into small quadratic problems. Decision table is similar to decision tree except that decision table needs consistent conditions to be evaluated as well as it needs consistent action that has to be done. Decision table generates rules from the data in the table, and its causes and effects are listed by the use of the matrix. *K*-star classifier is an instant-based classifier, and its instance depends upon the class of training instance. The class of the test instance is determined by the similarity with the training class. *K*-star classifier differs from instance-based learner; only *k*-star uses entropy to evaluate the distance between two instances, and the closest distance will be selected to assign the instance to the new class.

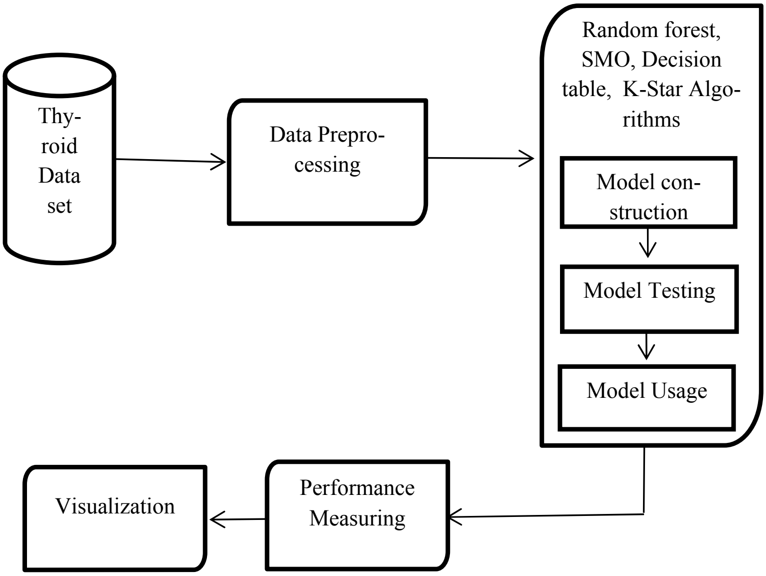


Fig. 1 Overview of the proposed work

These classification algorithms are used to build the classification model, test it, and then use the model for prediction of the disease. These classification algorithms predict hypothyroid disorder, and the performance of the model is analyzed by using K-fold cross-validation. It is observed that the random forest classification algorithm's performance is better than others, and it is compared with different algorithms by using WEKA tool. The detail about the classification performance of the model and the results obtained are shown in subsequent sections. The proposed model is illustrated in Fig. 1.

4 Result and Analysis

Hypothyroid data set is analyzed with different algorithms and different folds by using WEKA tool. Various performance metrics are used to determine the performance of the classification algorithms. Some of these parameters include accuracy rate, precision, recall, etc. The summarized result analysis of the classifiers used in our study with respect to classification accuracy rate is depicted in Table 1.

As it is observed from Table 1, random forest algorithm performs better when value of K is 6. Thus, using the K value as 6, performance analysis is carried out with other performance metrics like precision, recall, and F-measure. The results are shown in Table 2.

Table 1 Accuracy of each algorithm with different K values

Algorithm	$K = 2$ (%)	$K = 4$ (%)	$K = 6$ (%)	$K = 8$ (%)	$K = 10$ (%)
Random forest	98.83	99.47	99.44	99.47	99.31
Sequential minimal optimization	93.53	93.58	93.67	93.53	93.61
Decision rule	98.94	99.18	98.67	99.31	99.34
K-star	94.11	94.78	94.67	96.91	94.67

Table 2 Summarized result of each algorithm at $K = 6$

Algorithm	Random forest	SMO	Decision table	K-star
Accuracy	0.994	0.936	0.989	0.946
TP rate	0.785	0.500	0.755	0.500
Precision	0.968	0.941	0.972	0.676
Recall	0.785	0.500	0.755	0.579
F-measure	0.972	0.880	0.948	0.611

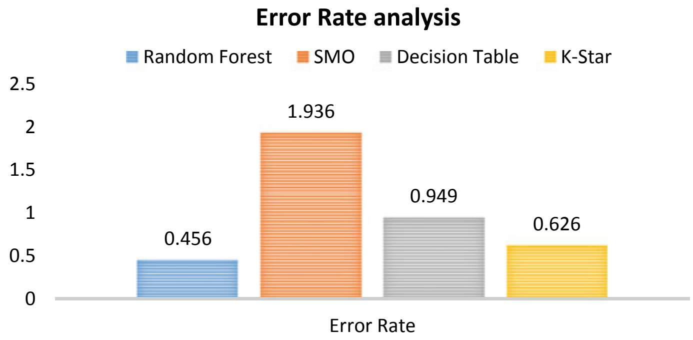


Fig. 2 Error rate comparison of classifiers under study

Figure 2 illustrates the graphical analysis of error rate of classification with different algorithms under study. The result clearly shows that error rate with random forest is minimum with 0.456.

5 Conclusion

Thyroid disorder is one of the main health problems which needs a special diagnosis. Data mining technique is applied to the hypothyroid data set, and data classification is to determine the positive and negative case. The experimental result shows that classification with random forest and decision table rule gives better accuracy with less error rate compared to SMO and K-star. At $K = 6$, random forest provides 99.44 percent accuracy, decision table 98.97, K-star gives 94.671, and SMO 93.673. At $K = 8$, random forest accuracy is 99.47 and decision table rule accuracy is 99.31. Generally, random forest classification and decision table rule have good performance for the prediction of thyroid disorder. Our future work constitutes an analysis of the classification performance of random forest algorithm on other critical health disorders.

References

1. Polat, K., Seral, S., Güneş, S.: A novel hybrid method based on artificial immune recognition system (AIRS) with fuzzy weighted pre-processing for thyroid disease diagnosis. Expert. Syst. Appl. **32**(4) (2007)
2. Keleş, A.: ESTDD: Expert system for thyroid diseases diagnosis. Expert. Syst. Appl. **34**(1) (2008)
3. Temurtas, F.: A comparative study on thyroid disease diagnosis using neural networks. Expert. Syst. Appl. **36**(1) (2009)
4. Dogantekin, E., Dogantekin, A., Avci, D.: An expert system based on generalized discriminant analysis and wavelet support vector machine for diagnosis of thyroid diseases. Expert Syst. Appl. **38**(1) (2011)
5. Chen, H. L., Bo, Y., Gang, W., Jie, L., Chen, Y. D., Liu, D. Y.:. A three-stage expert system based on support vector machines for thyroid disease diagnosis. J. Med. Syst. **36**(3) (2012)
6. Nikita, S., Alka, J.: A segmentation method and comparison of classification methods for thyroid ultrasound images. Int. J. Comput. Appl. **50**(11), 0975–8887 (2012)
7. Edgar, G.: Towards high-performance cell segmentation in multispectral fine needle aspiration cytology of thyroid lesions. Comput. Methods Programs Biomed.
8. Preeti, A.: Role of segmentation in medical imaging: A comparative study. Int. J. Comput. Appl. **29**(1), 0975–8887 (2011)
9. Restraints, G.: TND: A thyroid nodule detection system for analysis of ultrasound image and videos. Springer Science and Business Media, LLC (2010)
10. Dendi Maysanjaya, I. M., Hanung, A. N., Noor, A. S.: A comparison of classification methods on diagnosis of thyroid diseases. In: International Seminar on Intelligent Technology and Its Applications, IEEE (2015)

Global Best-Guided Modified Cat Swarm Optimization Algorithm for Applications in Machine Learning

Archana Sarangi, Shubhendu Kumar Sarangi, and Sonali Samal

Abstract The cat swarm optimization (CSO) is regarded as one of the intellectual optimization techniques that have the ability to show a good performance in the machine learning environment. The modified and improved version of the cat swarm optimization algorithm proposed in this paper provides a unique characteristic of parallel computation. The modified algorithm uses two types of weight expression to calculate the new weight for each iteration of the algorithm. The weight expression which helps in getting a better value of global best during the evaluation procedure is taken into consideration in every iteration. This modified algorithm takes the best from both the modified weight expression. Simulation results show that this modified version of CSO performs reasonably well in comparison to normal CSO.

Keywords Cat swarm optimization · Global best-guided modified cat swarm optimization · Benchmark functions

1 Introduction

Optimization of various parameters is the prime requirement of modern engineering applications. Due to the complicacy of modern world engineering and scientific problems, various types of bio-inspired computing techniques are used to determine the most preferred solutions. Swarm intelligence plays a vital role in the field of bio-inspired computing. The cat swarm optimization is generally recognized as one of the most demanded algorithms for optimization in the area of swarm computing. The most useful achievement of this algorithm is that it provides a simpler way

A. Sarangi · S. Samal
Department of Computer Science and Engineering, ITER, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India

S. K. Sarangi (✉)
Department of Electronics and Instrumentation Engineering, ITER, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: shubhendu1977@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_40

of programming while optimizing the complicated parameter values. A few modifications are done with the original algorithm to enhance its searching ability in exploration as well as exploitation. This paper presents research where an attempt has been made for providing an enhanced searching ability. In this research, two types of improved weight factors are used for updating the global best values in the solution. These two modified expressions of weight factor are evaluated concurrently during the execution of the algorithm. The improved weight factor which gives a better global best value in each iteration controlled the next iteration of the algorithm execution by acting as a reference. This process is repeated for controlling the movement of the algorithm in the best possible way. The performance of this new modified version is compared with the traditional cat swarm optimization algorithm by utilizing standard benchmarking functions. The sections following this introduction describe the details of the algorithm along with the analysis of results. The conclusion of an innovative modified version of the cat swarm algorithm is presented at the end of this paper.

2 Computing Techniques

2.1 Cat Swarm Optimization (CSO)

The two scientists Chu and Tsai projected the cat swarm optimization algorithm in 2007 [1] in the field of bio-inspired computing. This algorithm provides representation of the use of cats in two modes which are known as "Seeking mode" and "Tracing mode". In this algorithm, each searching particle represents a cat that is involved in locating the optimal values of various parameters. No. of cats generally represents as a number of particles for searching the solutions of the targeted problem. In this case, every cat is assigned a position with D number of dimensions where each dimension has a velocity component. The fitness assessment of the cats decides the regulation power of accommodation of the cat to the targeted fitness function. Generally, a flag is used to decide the involvement of the cat in seeking mode or tracing mode [2]. The best location attained by the cat at the end of the evaluation process is used as the final solution. CSO has two modes for optimization, i.e., seeking mode as well as tracing mode during the execution of the algorithm.

The update equations for position and velocity in each iteration are given by:

$$\mathrm{Vel}_{\mathrm{id}} = \mathrm{wt} * \mathrm{Vel}_{\mathrm{id}} + \mathrm{cc} * \mathrm{random} * (\mathrm{Pos}_{\mathrm{gd}} - X_{\mathrm{id}}) \tag{1}$$

$$X_{\mathrm{id}} = X_{\mathrm{id}} + \mathrm{Vel}_{\mathrm{id}} \tag{2}$$

where wt is the inertia weight, cc is the acceleration constant and random is a uniform distribution within the range [0, 1]. $\mathrm{Vel}_{\mathrm{id}}$ is the velocity of every cat and $\mathrm{Pos}_{\mathrm{gd}}$ global best location X_{id} is the position of every cat. The steps of the algorithm are described as follows:

- Assignment of position and velocity arbitrarily to L number of cats with D as dimension.
- Computation of the strength of each cat by storing the most excellent position.
- The mixture ratio along with the SMP decides the number of cats either involved in seeking mode or taking part in tracing mode.
- Using the flag, change the position of cats if the cat is involved in seeking mode, then seeking process is applied otherwise it goes for tracing mode.
- Evaluate the fitness of the cat again with storing of the best position.
- Compare local and global best positions for a better approach while updating the better position and update accordingly.
- If termination condition is fulfilled then stop to get finest results in solution otherwise repeat stops.

2.2 Global Best-Guided Modified Cat Swarm Optimization (GCSO)

In the normal cat swarm optimization algorithm, the convergence to the finest optimal value is difficult to achieve as the weight factor is not helping in getting a better convergence. The major limitation is that it is very difficult to get best values of weight factor as the execution progresses. This problem can be solved by the suitable selection of the weight factor by the use of modified expression. The weight factor in the modified expression is generally calculated in every iteration as the execution of the algorithm progress. This removes the disadvantages of assuming a fixed value of weight in the entire simulation. In this approach, a further modification is introduced by considering two values of modified weight expression [3]. The weighted calculation is not convergence to best values in all the cases in spite of using a modified expression. This limitation can be overcome by using two standard modified expressions for updating the weights while running the algorithm. But the major change is introduced in this case is the operation of both the weight updating sequentially and concurrently. The global best value is calculated by the employment of both the modified expression of cat swarm optimization with the use of weight expression as described below. This approach remarkably uses the best values obtained from both these modified weight expression without blindly depending on only one. These two improved weight expressions are used in the expression Eq. (1) after calculation is done for new value of weight from the current execution data.

The first weight factor used in the modified CSO algorithm is given by [4]

$$w(t) = w\text{start} * u^t$$

where u= [1.0001, 1.005]

Similarly, the second modified expression for the weight factor is also given by [5]

$$w(t) = \frac{t_{max} - t}{t_{max}}\left(w_{max} - w_{min}\right) + w_{min} * z$$

where t_{max} and t_{min} specify the maximum value of iteration along with t as the current iteration. w_{max} and w_{min} denote the maximum and minimum of the weight value within a range. The value of z specifies the random value taken randomly for better convergence. After utilizing the modified values of new generated weight, the global best position is calculated for each type of weight. The weight factor which gives better value is used in the next step of execution.

The execution sequence of the algorithm can be described as

- Assignment of position and velocity arbitrarily to L number of cats with D as dimension.
- Computation of the strength of each cat by storing the most excellent position.
- The mixture ratio along with the SMP decides the number of involved cats either taking part in seeking mode or in tracing mode.
- Using the flag, change position of cats if the cat is in seeking mode, then seeking procedure is applied otherwise it goes for tracing mode.
- Utilize the two modified improved expressions to calculate the next global best position. The weight expression which gives a better global best location is taken into consideration in the next iteration.
- Evaluate the fitness of the cat again with storing of the best position.
- Compare local and global best positions for a better approach while updating the better position and update accordingly.
- If termination condition is fulfilled then stop to get finest results in solution otherwise repeat stops.

3 Result Analysis

The suggested customized version of the existing cat swarm optimization algorithm can be evaluated by comparing it with the normal CSO algorithm. This procedure needs the use of several standardizing functions. The evaluation procedure involved the cat swarm optimization (CSO) and Global best-guided customized cat swarm optimization (GCSO). The objective behind the experiment is to lessen the function values utilized in the simulation. For convenience, the identical exploring ranges are defined for all assessment functions as $[-100{,}100]^D$. The simulation utilizes standard simulation parameters for verifying the performance of the proposed GCSO algorithm. The various specifications used in the simulations are specified follows. For Cat Swarm Optimization, Seeking Range of selected Dimension (SRD) = 0.2, Counts of Dimension to Change (CDC) = 0.8, Mixture Ratio (MR) = 0.9, Seeking Memory Pool (SMP) = 5, CC = 2, Population (L) = 100, Dimension (D) = 20, Max

Table 1 Benchmark functions

Sl.	Function name	Function expression
1	Griwank	$f(x') = \sum_{i=1}^{dd} \frac{x_i'^2}{4000} - \prod_{i=1}^{dd} \cos\left(\frac{x_i'}{\sqrt{i}}\right) + 1$
2	Katasura	$f(x') = \frac{10}{\mathrm{DD}^2} \prod_{i=1}^{\mathrm{DD}} \left(1 + i \sum_{j=1}^{32} \frac{\lvert 2^j x_i' - \mathrm{round}(2^j x_i')\rvert}{2^j}\right)^{10/\mathrm{DD}^{1.2}} - \frac{10}{\mathrm{DD}^2}$

Table 2 Performance analysis for function-1

Algorithm	Best	Worst	Mean
CSO	0.5692	0.6972	0.6425
GCSO	0.4344	0.5438	0.4706

Table 3 Performance analysis for function-2

Algorithm	Best	Worst	Mean
CSO	0.14287	0.15324	0.14316
GCSO	0.00346	0.00465	0.00381

Iteration = 200. For GCSO the parameters are taken as $w_{\min} = 0.4$ and $w_{\max} = 0.9$. All the experiments are repeated for 30 trials for checking the accuracy and precision of the projected algorithm. The normal mean fitness of the most excellent particles during the optimization runs has been recorded. In addition to that, a record was also made for the finest and worst values produced in all the function evaluations in the complete experiment. The benchmarking functions [6] utilized in this paper are listed in Table 1. The results of both the simulated standard functions are tabulated in Tables 2 and 3. The convergence profiles of both the functions simulated by all the four algorithms are described by Figs. 1, 2, 3, and 4.

The outcome of the simulating experiment utilizing normal cat swarm optimization algorithm with global best-guided modified cat swarm optimization algorithm is listed in Tables 2 and 3 respectively. Table 2 lists the finest, worst, and mean observations for the function one whereas Table 3 describes the finest, worst, and mean observed values for function 2, respectively. It can be easily visualized from the list of values depicted in the tables that the newly proposed Global best-guided modified cat swarm optimization performs in a better way after the evaluation is done with 30 independent runs. When compared with the normal cat swarm algorithm. The performance is well exhibited in terms of best, worst, and mean values. The best values produced in normal CSO for the standard functions utilized in the simulation are 0.5692 and 0.14287 respectively. But the new modified GCSO algorithm provides a better result than this standard CSO algorithm which is 0.4344 and 0.00346 consecutively. The mean values obtained all through the complete simulating experimentation for the new modified algorithm (GCSO) are found as 0.4706 and 0.00381 for functions 1 and 2 consecutively which shows the precision of the measurement by this algorithm. But the standard CSO provides mean values of

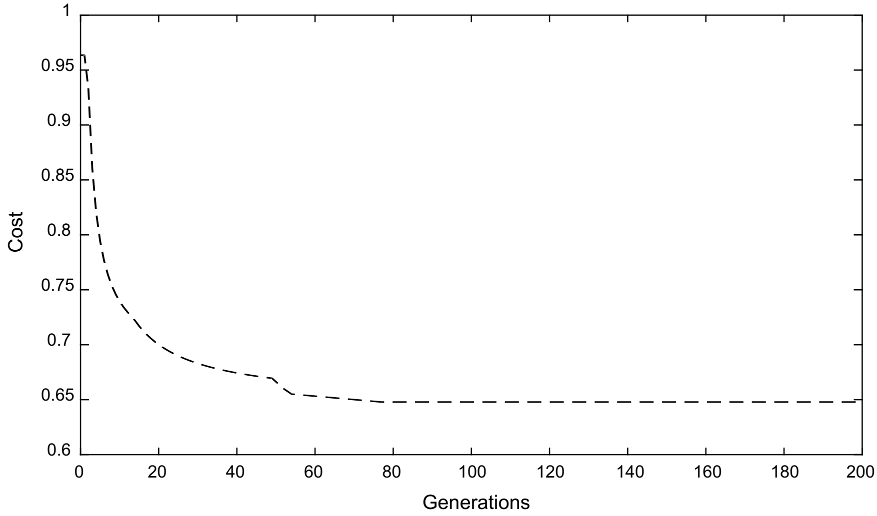


Fig. 1 Convergence profile for function-1 for CSO

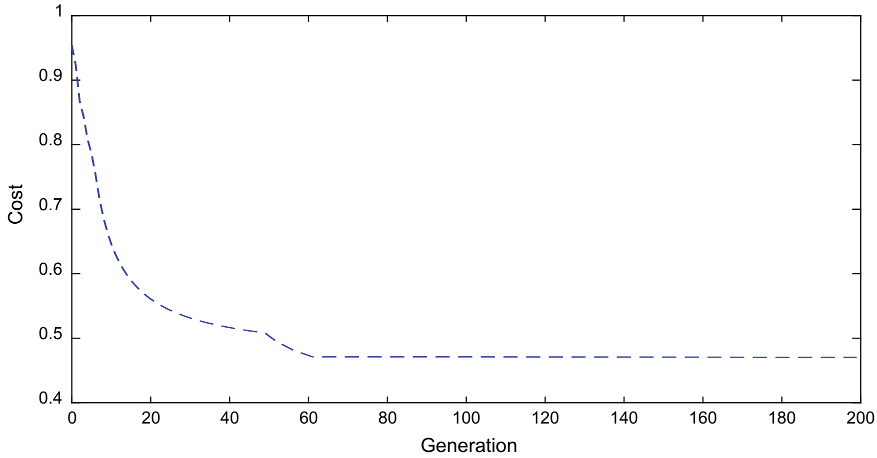


Fig. 2 Convergence profile for function-1 for GCSO

0.6425 and 0.14316, respectively, for functions 1 and 2 consecutively. The worst values obtained during the simulating experiment also agree with the fact that the proposal of a new improved version of the algorithm is fruitful. Figures 1 and 3 correspond to the convergence profile of the traditional cat swarm optimization algorithm for both functions whereas Fig. 2 as well as Fig. 4 correspond to convergence profile of the innovative customized Global best-guided cat swarm optimization algorithm for the same functions consecutively. In all the figures it can be noticeably established

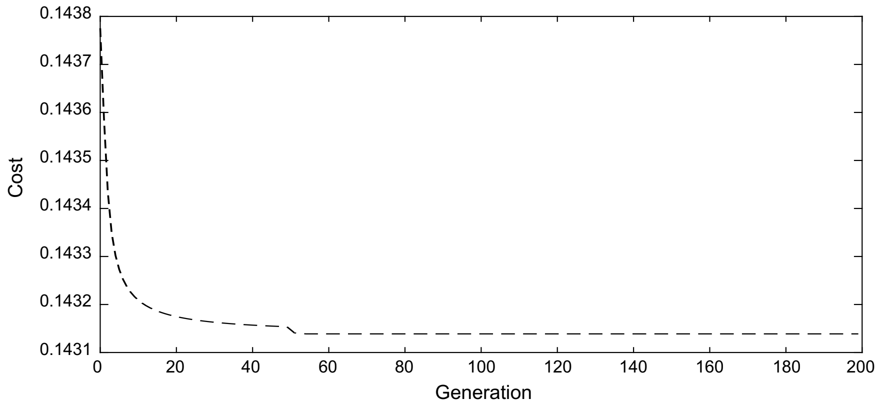


Fig. 3 Convergence profile for function-2 for CSO

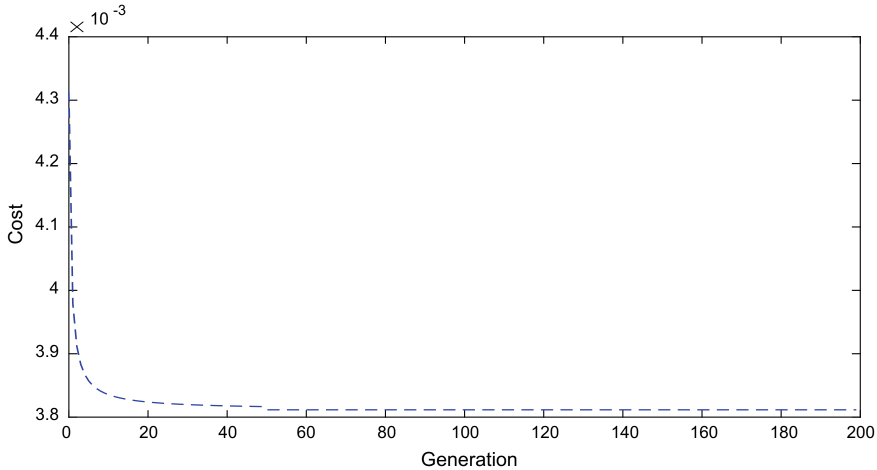


Fig. 4 Convergence profile for function-2 for GCSO

that the profile of convergence of the new customized algorithm is much enhanced when compared to earlier normal cat swarm optimization algorithm, i.e., CSO. It can also be visibly established that in almost all figures the new modified algorithm converges at a quicker rate in comparison to the prior existed standard algorithm. The performance of the projected algorithm is effective in both the unimodal as well as multi-modal problems of optimization. This experiment also verifies the effectiveness of the new modified algorithm is also better than standard CSO. The betterment in mean values obtained by the new modified algorithm also helps in compensation of time consumption in comparison to the normal CSO as the execution time of the novel modification is a little higher.

4 Conclusion

This paper proposed an upgraded version of the traditional generalized cat swarm algorithm in the field of computing inspired by the biological world. The improved modified version recognized as global best-guided cat swarm optimization which used two types of modified improved expression for a better approach to the solution. Both the improved weight expression is involved in the simulation during the evaluation of the algorithm. Both the global best values are calculated by each weight expression compared and the best value is taken into consideration. This approach removes the limitation of dependence on only one parameter thereby exploitation of the search space more deeply. This concurrence concept of upgrading weights is found suitable in GCSO algorithm is also found successful while doing the simulation. This proposal also very simple thereby not increases the complexity in the computation. The new modified algorithm provides improved exploitation of the search space. The algorithm proves its usefulness in the domain of standard statistical parameter values while the comparison is done with the normal foundation algorithm on experimenting utilizing standardized benchmarking functions. The superiority of the achieved results compensates for the little higher amount of execution time. Therefore, the new approach of modification of CSO can be used in a variety of machine learning applications for training purposes for getting improved results in real-world applications.

References

1. Chu, S.C., Tsai, P.W.: Computational intelligence based on the behavior of cats. Int. J. Innov. Comput. Inf. Control. **3**, 163–173 (2007)
2. Sarangi, A., Sarangi, S.K., Panigrahi, S.P.: An approach to identification of unknown IIR systems using crossover cat swarm optimization. Perspect. Sci. **8**, 301–303 (2016)
3. Tian, D., Shi, Z.: MPSO: Modified particle swarm optimization and its applications. Swarm Evol. Comput. **41**, 49–68 (2018)
4. Jiao, B., Lian, Z., Gu, X.: A dynamic inertia weight particle swarm optimization algorithm. Chao Solit Fractals **37**, 698–705 (2008)
5. Feng, Y., Yao, Y., Wang, A.: Comparing with chaotic inertia weights in particle swam optimization. In: Proceedings of the International Conference on Machine Learning and Cybernetics, pp. 329–333 (2007)
6. Chen, Q., Liu, B., Zhang, Q., Liang, J. J., Suganthan, P. N., Qu, B. Y: Problem definitions and evaluation criteria for CEC 2015 special session on computationally expensive single objective optimization, Technical Report (2014)

Analysis of Prediction Accuracy of Diabetes Using Classifier and Hybrid Machine Learning Techniques

Shekharesh Barik, Sambit Mohanty, Surajit Mohanty, and Debabrata Singh

Abstract In the past few years, the growth of diabetes among people became exponential. A health report tells that about 347 million of world populations are affected by diabetes. Diabetes not only affects the older person but the younger generation too. To detect diabetes at an early stage is also a big challenge. This detection will be helpful for decision-making process of medical system. Early prediction of diabetes helps us to save the human life from diabetes. A prolong diabetes leads to the risk of damage in vital organs of human body. So, early prediction of diabetes is very crucial in order to save human being from diabetes. Data analysis is concerned with finding a pattern from a large dataset. This helps us to build certain conclusion out of the available datasets. The analytical process can be done by different machine learning algorithms. This paper presents two sets of machine learning approach for prediction of diabetes. One of them is a classification-based algorithm, and the other one is a hybrid algorithm. In classification, we have taken the random forest algorithm. For hybrid approach, we have chosen XGBoost algorithm. These two algorithms were implemented and compared in order to explore the prediction accuracy in diabetes for two different machine learning approaches and got the mean score 74.10% which is better than the Random Forest algorithm.

S. Barik · S. Mohanty
Department of Computer Science and Engineering, DRIEMS (Autonomous), Cuttack, Odisha, India
e-mail: shekharesh@gmail.com

S. Mohanty
e-mail: mohanty.surajit@gmail.com

S. Mohanty
SLFS Lab, Bhubaneswar, Odisha, India
e-mail: sambitmohanty778@gmail.com

D. Singh (✉)
Department of Computer Science and Information Technology, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: debabratasingh@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_41

Keywords Decision-making system · Data analysis · Hybrid algorithm · Random forest · XGBoost

1 Introduction

Diabetes arises when sugar level in blood increases. Nowadays, diabetes is a fast-growing disease. Most physician suggests that the main factors for diabetes are bad life style, bad diet, and lack of exercise. The other factors can be obesity, viral infection, chemical in food, environment pollution, immune reaction, bad food habit, i.e., more oil and salt intake in food. Diabetes can be classified into three categories: Type 1, Type 2, and Type 3 [1]. In Type 1 diabetes, the immune system of body destroys the important cells which produce insulin to absorb the sugar. Type 1 diabetes can attack child or adult. Type 2 diabetes is found in adults, mainly in middle aged or old aged people. Obesity is one of the causes for Type 2 diabetes; obesity is the imbalance of body mass index (BMI) [2]. In Type 2 diabetes, the human body is unable to produce insulin [3]. Type 3 diabetes is also known as gestational diabetes. This type of diabetes is mainly found in case of pregnant women. They develop high glucose level in blood, which needs urgent medical attention. Otherwise, it will lead to various complications during pregnancy. If we neglect toward diabetes, then it can affect and damage the vital organs of body such as heart, liver, kidney, and eyes. Apart from that, the progression of diabetes occurs through five stages except Stage 0 which is considered as normal [4, 5]. Stage 1: Sometimes called compensation where secretion of insulin increases to maintain normoglycemia and β cell mass decreases. Stage 2: Glucose levels start increasing and β cell mass also decreases. Stage 3: It is an unstable period in which glucose levels relatively increase and approach to Stage 4. Stage 4: It is characterized as stable decompensation, and β cell is more dedifferentiation. Stage 5: Severe decompensation occurs, and β cell mass severely reduces and progression to ketosis as depicted in Fig. 1a, b.

Data analysis deals with identifying a pattern from a large set of data. From these analysis, we can derive certain predictive conclusions. This analytical work can be accomplished by the machine learning algorithms [6]. Machine learning is a part of artificial intelligence. Like human, brain learns from the past experience or analyzing the past history. Machine also learns in the similar way. Hence, machine can take its decision depending upon the knowledge fed into it. Knowledge is gathered by analyzing huge amount of data and by providing training to machine. We are using the machine learning algorithms: random forest classifier and XGBoost. We have implemented these two algorithms using a medical dataset to explore their techniques. These algorithms belong to two different machine learning approaches. So, we can get a thoughtful insight and conclusive remark regarding the approaches by comparing the prediction results [7, 8].

The remaining part of this research paper is organized as follows: Section 2 briefly represents the literature survey of various techniques for prediction of the diabetes. Section 3 presents the methodology used, i.e., Random forest classifier and XGBoost

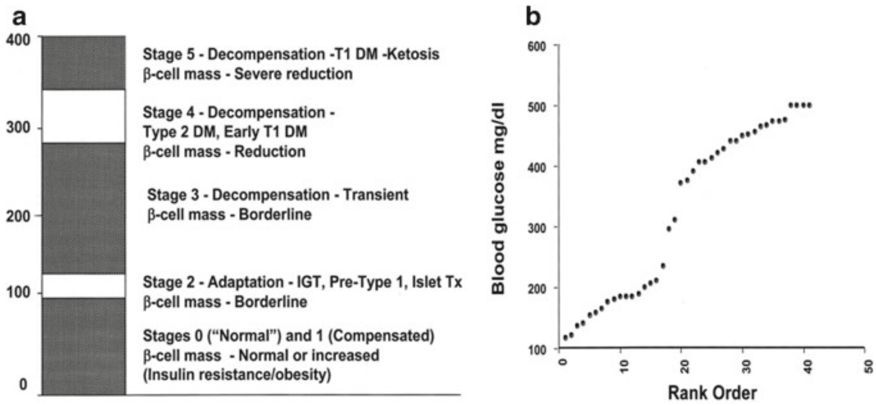


Fig. 1 **a** Five stages of progression of diabetes. **b** Normal and severe diabetic sugar levels

(hybrid machine learning technique) with the considered dataset. In Sect. 4, we compared the two methodologies and discussed the result analysis elaborately, and finally in Sect. 5 we conclude our paper with future work and some references.

2 Related Works

Sisodia et al. in [9] predict the diabetes using machine learning approaches. Using classification algorithm, i.e., decision tree, SVM, and Naïve Bayes, the authors detect diabetes at an early stage. From the result analysis, authors show that Naïve Bayes gives 76.3% highest accuracy comparatively with other approaches. Aljumahi et al. in [10] predict the diabetes treatment using the deterministic regression-based techniques. They adopted Oracle Data Miner (ODM) for predicting modes of treating diabetes into two groups, i.e., young and old age groups. From their analysis, young group can be delayed to avoid side effect, whereas in old age group, prescribed drug should be immediately taken. Lyer in [11] find out one of the finest solutions to diagnose the diabetic disease through the patterns found by the help of data classification analysis on J48 decision tree and Naïve Bayes algorithm. From their analysis, Naïve Bayes techniques give least error rate as compared to the other approaches.

Velu in [12] employed the most emerged three techniques for classification of the diabetic patients, i.e., EM algorithms, H Means + clustering, and Genetic Algorithm (GA) [6]. From their result analysis, H Means + clustering techniques give a better result as compared to other two techniques in case of diabetes disease. Ganji in [13] adopted fuzzy ant colony optimization techniques to find the set of rules for the adiabatic patient and their diagnosis. Now it is also used for the prima Indian diabetes datasets. Jayalakshmi T. in [14] diagnoses the adiabatic patient through their new approach—ANN techniques. The authors preprocessed and replaced the missing values in the datasets used for detecting the diabetic patient. Their modification on dataset gives a better accuracy as compared to other training datasets, due to getting result in lesser time. Aishwarya et al. in [15] used classification techniques by using machine learning approaches for diabetes. To detect diabetes disease at an early stage, a greatest support of machine learning is needed. Authors trying a promising technique support vector machine (SVM) in machine learning approaches for classification.

3 Methodology Used

In this paper, we have classified the patient data to predict whether a patient has diabetes or not. For this classification purpose, we have used PIMA diabetes dataset which is provided by National Institute of Diabetes (NID) for Digestive and Kidney Diseases. This dataset consists of 768 rows and 9 columns. Each row has nine attributes like (i) Pregnancies (Number of times pregnant), (ii) Glucose (Plasma glucose concentration within 2 h duration with an oral glucose tolerance test), (iii) Blood Pressure [Diastolic/Systolic blood pressure (mm Hg)], (iv) The Skin Thickness [Triceps skin fold thickness (mm)], (v) Insulin [2-h duration serum insulin (mu U/ml)], (vi) BMI [body mass index (weight in kg/(height in m)^2)], (vii) Diabetes_Pedigree_Function (Diabetes pedigree function), (viii) Age[(years)], and (ix) Outcomes [Class variable (0 or 1)] [16]. These columns indicate some specific medical conditions, and the snapshot of the datasets is given below:

Out[4]:

	num_preg	glucose_conc	diastolic_bp	thickness	insulin	bmi	diab_pred	age	skin	diabetes
0	6	148	72	35	0	33.6	0.627	50	1.3790	True
1	1	85	66	29	0	26.6	0.351	31	1.1426	False
2	8	183	64	0	0	23.3	0.672	32	0.0000	True
3	1	89	66	23	94	28.1	0.167	21	0.9062	False
4	0	137	40	35	168	43.1	2.288	33	1.3790	True

Here the outcome column has two values, 0 and 1. 0 means patient has no diabetes, and 1 means patient has diabetes. Within this paper, we have used two algorithms like RandomForest classifier and XGBoost algorithm to predict whether a patient has diabetes or not [17].

We have used Jupyter Notebook which is freely available software for performing machine learning operations. For machine learning purpose, we need to import the Sklearn module which contains all the essential algorithm and functions. We need to import Python NumPy module and Pandas module for data analysis purpose. To plot different graphs, we need to import Matplotlib module which contains all the methods related to plotting graphs. Our data is stored in a CSV file which needs to be imported to the notebook by using Python Pandas module as a data frame. After importing the data, we can apply various data analysis and machine learning algorithms for classification and prediction. Before applying the machine learning algorithms on the dataset, we can see it through graphs by using Matplotlib.

If we want to see the correlation among the different parameters of the dataset, then we need to use the corr() function, which will find the correlation among the variables. Then, we can plot this correlation values in the form of graph by using seaborn module. To see this correlation properly, we have to use the heat map as shown below that also shows the correlation among the attributes of the datasets depicted in Fig. 2. The correlation values for the dataset are as follows:

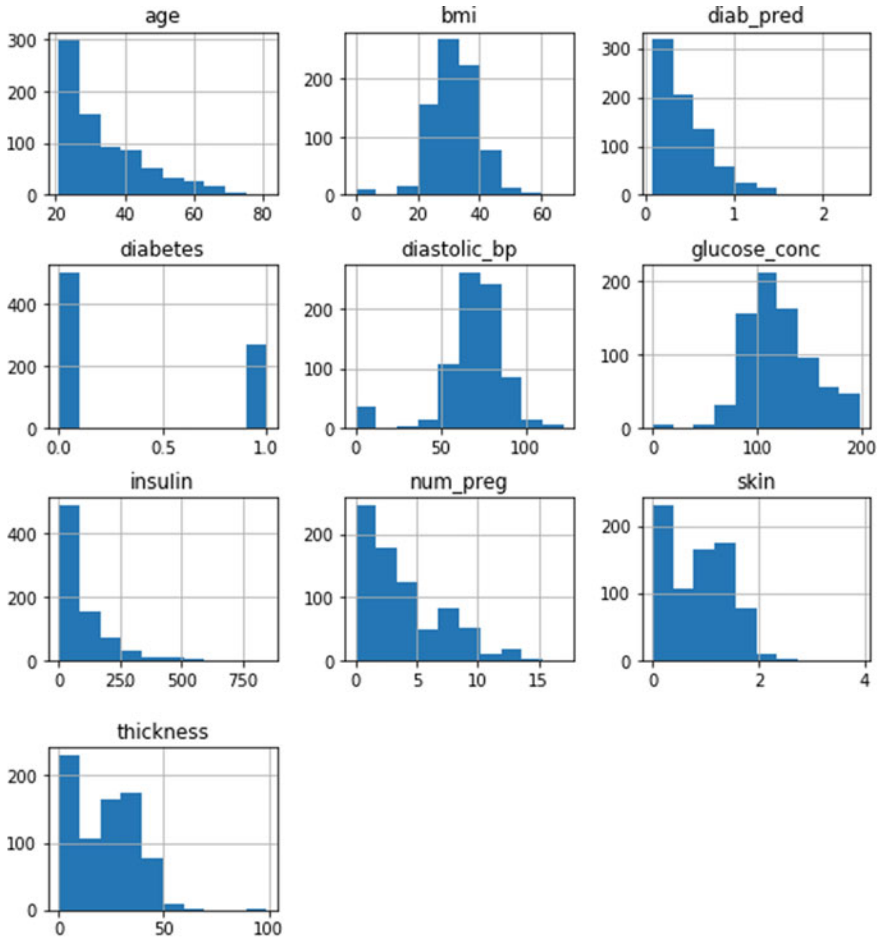


Fig. 2 Features/attributes of the dataset used in our classification

In [8]:
```
data.corr()
```

Out[8]:

	num_preg	glucose_conc	diastolic_bp	thickness	insulin	bmi	diab_pred	age	skin	diabetes
num_preg	1.000000	0.129459	0.141282	-0.081672	-0.073535	0.017683	-0.033523	0.544341	-0.081672	0.221898
glucose_conc	0.129459	1.000000	0.152590	0.057328	0.331357	0.221071	0.137337	0.263514	0.057328	0.466581
diastolic_bp	0.141282	0.152590	1.000000	0.207371	0.088933	0.281805	0.041265	0.239528	0.207371	0.065068
thickness	-0.081672	0.057328	0.207371	1.000000	0.436783	0.392573	0.183928	-0.113970	1.000000	0.074752
insulin	-0.073535	0.331357	0.088933	0.436783	1.000000	0.197859	0.185071	-0.042163	0.436783	0.130548
bmi	0.017683	0.221071	0.281805	0.392573	0.197859	1.000000	0.140647	0.036242	0.392573	0.292695
diab_pred	-0.033523	0.137337	0.041265	0.183928	0.185071	0.140647	1.000000	0.033561	0.183928	0.173844
age	0.544341	0.263514	0.239528	-0.113970	-0.042163	0.036242	0.033561	1.000000	-0.113970	0.238356
skin	-0.081672	0.057328	0.207371	1.000000	0.436783	0.392573	0.183928	-0.113970	1.000000	0.074752
diabetes	0.221898	0.466581	0.065068	0.074752	0.130548	0.292695	0.173844	0.238356	0.074752	1.000000

3.1 Random Forest Classifier

Random forest belongs to the category of supervised machine learning. Forest refers to a set of trees, and tree refers to the decision tree. The prediction value of random forest depends on the decision trees present in the forest. Each decision tree represents some feature or label of dataset on random basis. Decision tree gives the prediction value after performing the operation on given data. Voting is done in case of classification problems. Majority in voting decides the class to be chosen [18]. Random forest can also be applied for regression-based problems. Here, the output is found by calculating the mean or median of all the predicted values given by the decision trees [19]. One of the main problems in decision tree is the high variance and low bias. Random forest overcomes this problem with many number of decision trees. This also gives us high degree of prediction accuracy [20, 21]. That is why random forest is a very popular machine learning algorithm. It can be applied for disease prediction, credit card fraud detection, design of recommendation system, classification of loan applicants, and so on. One of the challenges in random forest may be slower prediction time. This is because random forest uses many decision trees [17, 22].

In order to use Random forest classifier on our dataset, we have to import the RandomForestClassifier class from Sklearn module. We need to apply some exploratory data analysis using pandas and NumPy before applying RandomForest algorithm on the dataset. We have to find how many zero values are present in all the columns of the dataset except the Pregnancy column as its medical value is taken 0 or any other integer values basing on number of times patient is pregnant. But for all other columns, some values must be recorded that need to be checked. After applying exploratory data analysis, we got the results shown below:

```
total number of rows : 768
number of rows missing glucose_conc: 5
number of rows missing glucose_conc: 5
number of rows missing diastolic_bp: 35
number of rows missing insulin: 374
number of rows missing bmi: 11
number of rows missing diab_pred: 0
number of rows missing age: 0
number of rows missing skin: 227
```

We need to change these 0 values or missing values to some numerical value. For this purpose, we can either take mean or median value as a replacement. In this case, we have taken mean value to replace these missing values. After replacing all missing values with mean value, we need to divide the dataset into train and test parts. Here, 30% of total dataset is taken for testing purpose, and 70% is taken for training.

```
In [15]: ## Train Test Split

         from sklearn.model_selection import train_test_split
         feature_columns = ['num_preg', 'glucose_conc', 'diastolic_bp', 'insulin', 'bmi', 'diab_pred', 'age', 'skin']
         predicted_class = ['diabetes']

In [16]: X = data[feature_columns].values
         y = data[predicted_class].values


         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.30, random_state=10)
```

After dividing dataset into train and test, now we can apply Random Forest algorithm on the train dataset for training the model.

```
In [19]: ## Apply Algorithm

         from sklearn.ensemble import RandomForestClassifier
         random_forest_model = RandomForestClassifier(random_state=10)

         random_forest_model.fit(X_train, y_train.ravel())
```

Now the model is trained by learning from the train dataset, i.e., X_train. Now we have to test our model by using the test dataset which we have separated from the original dataset.

```
In [23]: predict_train_data = random_forest_model.predict(X_test)

         from sklearn import metrics

         print("Accuracy = {0:.3f}".format(metrics.accuracy_score(y_test, predict_train_data)))

         Accuracy = 0.719
```

When we used our test dataset, i.e., X_test, it gives accuracy of 71.9%. The accuracy can be further improved by hyper-parameterization.

3.2 XGBoost

XGBoost is a type of hybrid machine learning algorithm which always provides better solution than any other machine learning algorithm. XGBoost or Extreme Gradient Boosting is a type of boosting algorithm based on ensemble. It combines several weak learners and provides an improve prediction accuracy. It builds weak models and understands various important features, parameters; using those conclusions, it builds a new stronger model and tries to reduce the rate of misclassification. It is also called gradient boosting because when we add the new models, it uses a gradient descent algorithm to minimize the loss. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. XGBoost uses a tree ensemble [16, 18]. Tree ensemble is a set of classification and regression trees. These trees try to reduce the misclassification rates on each iteration. XGBoost can be used to solve any kind of machine learning problems like regression, classification, ranking, and user-defined prediction problems. It supports various cloud platforms like AWS, Azure, and so on. It is a type of algorithm which uses less computing resources to provide the better accuracy in less time. It provides better results than other algorithms because it has some inbuilt features like parallel tree building, tree pruning using depth-first approach, and cache awareness by using internal buffers to store data, and uses regularization to avoid over-fitting, efficient handling of missing data, and inbuilt cross validation capability. XGBoost provides some kind of tuning parameters which must be optimized for better performance as discussed below:

i. Learning_rate: step size shrinkage used to prevent over-fitting. Range is [0, 1].
ii. Max_depth: determines how deeply each tree is allowed to grow during any boosting round.
iii. Subsample: percentage of samples used per tree. Low value can lead to under-fitting.
iv. Gamma: controls whether a given node will split based on the expected reduction in loss after the split. A higher value leads to fewer splits. It supports only the tree-based learners.
v. Colsample_bytree: percentage of features used per tree. High value can lead to over-fitting.
vi. N_estimators: number of trees you want to build.
vii. Alpha: L1 regularization on leaf weights. A large value leads to more regularization.
viii. Lambda: L2 regularization on leaf weights is smoother than L1 regularization.

```
In [26]: ## Hyper Parameter Optimization

         params={
          "learning_rate"    : [0.05, 0.10, 0.15, 0.20, 0.25, 0.30 ] ,
          "max_depth"        : [ 3, 4, 5, 6, 8, 10, 12, 15],
          "min_child_weight" : [ 1, 3, 5, 7 ],
          "gamma"            : [ 0.0, 0.1, 0.2 , 0.3, 0.4 ],
          "colsample_bytree" : [ 0.3, 0.4, 0.5 , 0.7 ]

         }
```

We have used RandomizedSearchCv to optimize the hyperparameter for XGBoost.

```
In [27]: ## Hyperparameter optimization using RandomizedSearchCV
         from sklearn.model_selection import RandomizedSearchCV
         import xgboost

In [28]: classifier=xgboost.XGBClassifier()

In [29]: random_search=RandomizedSearchCV(classifier,param_distributions=params,n_iter=5,scoring='roc_auc',n_jobs=-1,cv=5,verbose=3)
```

We need to predict the best parameter for XGBoost algorithm using Randomized-SearchCv.

```
In [34]: random_search.best_estimator_

Out[34]: XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                colsample_bytree=0.3, gamma=0.0, learning_rate=0.25,
                max_delta_step=0, max_depth=3, min_child_weight=7, missing=None,
                n_estimators=100, n_jobs=1, nthread=None,
                objective='binary:logistic', random_state=0, reg_alpha=0,
                reg_lambda=1, scale_pos_weight=1, seed=None, silent=True,
                subsample=1)

In [36]: classifier=xgboost.XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                colsample_bytree=0.3, gamma=0.0, learning_rate=0.25,
                max_delta_step=0, max_depth=3, min_child_weight=7, missing=None,
                n_estimators=100, n_jobs=1, nthread=None,
                objective='binary:logistic', random_state=0, reg_alpha=0,
                reg_lambda=1, scale_pos_weight=1, seed=None, silent=True,
                subsample=1)
```

Now we have got our best classifier as per our problem, we can now apply the model on the dataset.

```
In [39]: from sklearn.model_selection import cross_val_score
         score=cross_val_score(classifier,X,y.ravel(),cv=10)

In [40]: score

Out[40]: array([0.72727273, 0.77922078, 0.67532468, 0.67532468, 0.7012987 ,
                0.74025974, 0.76623377, 0.76623377, 0.77631579, 0.80263158])

In [41]: score.mean()

Out[41]: 0.7410116199589883
```

We got the mean score, i.e., 74.10%, which is better than the RandomForest algorithm. This model can be improved by applying some other optimization method. We can deploy our model to cloud hosting services as cloud services are providing

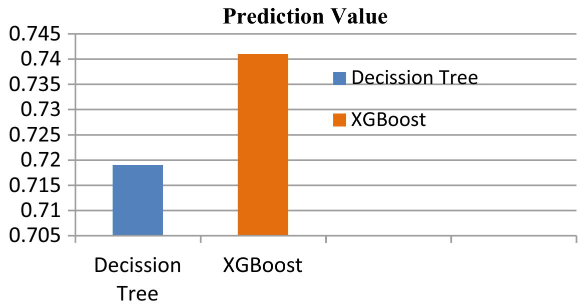


Fig. 3 Comparison of prediction value for machine learning algorithms

support for machine learning models which may not be supported by physical server. We can choose best cloud service provider as per our requirement. Some top most cloud service providers are Amazon, Google, Microsoft, and so on.

4 Result Analysis

We have used two machine learning algorithms on our dataset to get the prediction for diabetes. In case of Random forest with a fixed value of n, we get prediction value 0.719. But for XGBoost method, we get the prediction value 0.7410116. Hence, it is clear that XGBoost model gives more accuracy in our case (Fig. 3).

5 Conclusion and Future Scope

In this paper, we have applied two machine learning algorithms on the dataset for prediction of diabetes. RandomForest uses sequential decision trees, whereas XGBoost uses parallel trees for prediction. XGBoost provides better results and also faster than RandomForest as it optimally uses both hardware and software. These algorithms belong to two different machine learning approaches. More analytical report and concluding remark can be drawn by considering more machine learning algorithms which fall under these two approaches. We can also improve the performance of these algorithms by using hyper-parameterization and optimization methods which are beyond the scope of the paper. Furthermore, this work can be extended and improved by including other sets of machine learning approaches.

References

1. American Diabetes Association: Diagnosis and classification of diabetes mellitus. Diabet. Care **27**, S5 (2004)
2. Rasouli, B., Ahlbom, A., Andersson, T., Grill, V., Midthjell, K., Olsson, L., Carlsson, S.: Alcohol consumption is associated with reduced risk of type 2 diabetes and autoimmune diabetes in adults: results from the Nord Trøndelag health study. Diabet. Med. **30**(1), 56–64 (2013)
3. Singla, R., Singla, A., Gupta, Y.: Artificial intelligence/machine learning in diabetes care. Indian J. Endocrinol. Metabol. **23**(4), 495–497 (2019)
4. Jayalakshmi, T., Santhakumaran, A.: A novel classification method for diagnosis of diabetes mellitus using artificial neural networks. In: International Conference on Data Storage and Data Engineering, pp. 159–163 (2010)
5. Weir, G.C., Bonner-Weir, S.: Five stages of evolving beta-cell dysfunction during progression to diabetes. Diabetes **53**(Suppl 3), S16–S21 (2004)
6. Dash, R., Misra, B.B.: A multi-objective feature selection and classifier ensemble technique for microarray data analysis. Int. J. Data Min. Bioinform. **20**(2), 123–160 (2018)
7. Zheng, T., Xie, W., Xu, L.: A machine learning-based framework to identify type 2 diabetes through electronic health records. Int. J. Med. Inf. **97**, 120–127 (2017)
8. Nabi, M., Kumar, P., Wahid, A.: Performance analysis of classification algorithms in predicting diabetes. Int. J. Adv. Res. Comput. Sci. **8**(3), 456–461 (2017)
9. Sisodia, D., Sisodia, D.S.: Prediction of diabetes using classification algorithms. Proc. Comput. Sci. **132**, 1578–1585 (2018)
10. Aljumah, A.A., Ahamad, M.G., Siddiqui, M.K.: Application of data mining: diabetes health care in young and old patients. J. King Saud Univ. Comput. Inf. Sci. **25**(2), 127–136 (2013)
11. Iyer, A., Jeyalatha, S., Sumbaly, R.: Diagnosis of diabetes using classification mining techniques. arXiv preprint arXiv:1502.03774
12. Velu, C.M., Kashwan, K.R.: Visual data mining techniques for classification of diabetic patients. In: 2013 3rd IEEE International Advance Computing Conference (IACC), pp. 1070–1075. IEEE (2013)
13. Ganji, M.F., Abadeh, M.S.: Using fuzzy ant colony optimization for diagnosis of diabetes disease. In: 2010 18th Iranian Conference on Electrical Engineering, pp. 501–505. IEEE (2010)
14. Jayalakshmi, T., Santhakumaran, A.: A novel classification method for diagnosis of diabetes mellitus using artificial neural networks. In: 2010 International Conference on Data Storage and Data Engineering, pp. 159–163. IEEE (2010)
15. Aishwarya, R., Gayathri, P.: A method for classification using machine learning technique for diabetes (2013)
16. Mohanty, S., Sahoo J., Pramanik J., Jayanhu S.: A review of techniques in practice for sensing ground vibration due to blasting in open cast mining. In: International Conference on Remote Engineering and Virtual Instrumentation, pp. 306–314 (2019)
17. Aishwarya, R., Gayathri, P., Jaisankar, N.: A method for classification using machine learning technique for diabetes. Int. J. Eng. Technol. **5**, 2903–2908 (2013)
18. Rashid, T.A., Abdulla, S.M., Abdulla, R.M.: Decision support system for diabetes mellitus through machine learning techniques. Int. J. Adv. Comput. Sci. Appl. **7**, 170–178 (2016)
19. Wang N, Kang G (2012) Monitoring system for type 2 diabetes mellitus. In: IEEE Conference on E-health Networking, pp. 62–67
20. Kononenko, I.: Machine learning for medical diagnosis: history, state of the art and perspective. Artifi. Intell. Med. **23**(1), 89–109 (2001). (Elsevier)
21. Schulze, M.B., Hu, F.B.: Dietary patterns and risk of hypertension, type 2 diabetes mellitus, and coronary heart disease. Curr. Atherosclerosis Rep. **4**(6), 462–467 (2002)
22. Verikas, A., Gelzinis, A., Bacauskiene, M.: Mining data with random forests: a survey and results of new tests. J. Pattern Recogn. **44**, 330–349 (2011)

A Legendre Neural Network for Credit Card Fraud Detection

Rajashree Dash, Rasmita Rautray, and Rasmita Dash

Abstract Every year heavy financial losses are caused due to fraudulent activities in credit card transactions. These losses can be reduced through the development of efficient fraud detection algorithm. However, most transaction data are highly non-stationary and have unbalanced class distributions. Hence, the design of efficient fraud detection algorithm is always a challenging task for the fraud investigators. In this paper, a credit card fraud detection (CCFD) framework is proposed using a Legendre neural network (LENN). The model is validated by analyzing its performance on two credit card datasets. Result analysis clearly illustrates the better performance of LENN compared to a Chebyshev functional link artificial neural network (CHFLANN), multilayer perceptron (MLP) and a decision tree (DT)-based fraud detection framework.

Keywords Fraud detection · ANN · LENN · CHFLANN

1 Introduction

The rapid growth of online banking and e-commerce has opened up the path for the fraudsters to commit different types of financial frauds. Every year credit card fraud alone is reporting a huge loss in the revenue of the country. The main reason behind this is the rapid increase in online merchants, card users and issuers without involving more technology in detection and prevention of these frauds. Traditional approaches of fraud detection are leaned on manual techniques such as auditing. But

R. Dash (✉) · R. Rautray · R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: rajashree_dash@yahoo.co.in

R. Rautray
e-mail: rasmitarautray@yahoo.co.in

R. Dash
e-mail: rasmita02@yahoo.co.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_42

these techniques fail to provide efficient and reliable output for highly voluminous and unbalanced real-world data with dynamic behavior of the fraudsters. In literature, many researchers have suggested numerous data mining techniques to dissect fraudulent data from authentic one, thereby revealing fraudulent activities and facilitating decision-makers to develop well-planned tactics to reduce the impact of fraud.

In [1], the authors have presented a review that clearly shows an extensive application of six data mining techniques such as clustering, regression, classification, outlier detection, prediction and visualization for four types of monetary extortion that appeared in commodities, securities and bank transactions. A thorough review on types of fraud investigated, computational intelligence-based detection methods and performances of those methods for a couple of fraud types are presented in [2]. In [3], the performance of random forest (RF), logistic regression (LR) and support vector machine (SVM) are examined for detecting the fraud in a real-life credit card transactions collected during January 2006 to January 2007 from an international credit card operation. Various derived attributes are created in the study by aggregating the transaction data containing both numerical and categorical attributes. From experimental results, RF is found to surpass the other two techniques. A genetic algorithm combined with scatter search approach is analyzed in [4] for CCFD. In [5], examining two data mining techniques such as ANN and LR for CCFD clearly shows that ANN outperforms LR. In [6], the performance of a cost-sensitive DT is analyzed for credit card fraud detection. The model has shown better result compared to DT, ANN and SVM. A novel approach combining network-based and intrinsic features is suggested in [7] for identifying fishy credit card activities wielding in online stores. The fundamentals of RFM are used here to derive the intrinsic features from spending history of customer and characteristics of incoming transactions. A time-dependent score estimated for each network of credit card holders and merchants is taken as the network feature for the analysis. Testing the proposed approach on a RF, LR and NN model over a dataset containing around three million transactions, RF is found to be a best performing model leading a high accuracy. An innovative approach using self-organization map (SOM) is proposed in [8] to unravel, refine and analyze customer behavior for detecting fraud. A fraud detection model developed using artificial immune systems (AIS) are introduced in [9]. By adding improvements to the base AIS, the detection accuracy has appeared to be increased, thereby decreasing the cost and response time. In [10], a hybrid fraud detection model is presented combining fuzzy clustering with neural network. The model works in three phases. Following the initial step of checking user authentication and verification of card details, a fuzzy *c*-means clustering algorithm is applied to categorize the transactions as fraudulent or legitimate or suspicious from the normal usage patterns of credit card users rest on their past history. Finally, whether it is really a fraud transaction or an occasional deviation by a genuine user is detected by a neural network. In [11], a novel profit-based neural network is detecting the frauds with better accuracy compared to cost-sensitive ANN (CNN), DT and Naïve Bayes models. The study has includes both profit and accuracy-based performance metrics for doing analysis. In [12], three classifiers such as CHFLANN, MLP and DT are designed for addressing the credit card fraud detection problem. Comparing the three models with respect to classification accuracy and elapsed time, MLP is found to be better than CHFLANN and DT. Though a variety of data mining methods are explored in past, yet efficient fraud detection with real-world data persists as a major challenge.

In this paper, the performance of a LENN is analyzed for detecting fraud in credit card transactions. In literature, a number of applications of LENN are found in prediction purpose [13–16]. In this study, the network is designed as a classification model. An extensive simulation study showing the effectiveness of the model compared to other classifiers used for same purpose such as CHFLANN, MLP and DT over two credit card datasets is presented in the paper.

Rest of the paper is organized as follows: Sect. 2 describes the details of LENN-based fraud detection model. The detail of experimental result analysis is represented in Sect. 3 with a concluding report in Sect. 4.

2 LENN-Based Fraud Detection

This section outlines the working principle of LENN and the details of developing a classification model using the LENN for credit card fraud detection.

2.1 Legendre Neural Network

Legendre neural network (LENN) is one type of neural network, in which expansion blocks using Legendre orthogonal functions are used replacing the hidden layers with several neurons as in MLP. The expansion block helps to augment the dimension of the input pattern through a specified expansion order, which in turn helps to unravel the inherent nonlinearity exist between the original input and its corresponding output pattern. The simple structure of the network including a learning component and an expansion block results in less computational complexity and training time compared to a MLP with hidden layers [13–16].

The recursive formula to produce Legendre polynomials of higher order is given as follows:

$$\mathrm{LE}_{m+1}(s) = 1/(m+1)[(2m+1)s\mathrm{LE}_m(s) - m\mathrm{LE}_{m-1}(s)] \tag{1}$$

where m represents the expansion order, s is the input sample. $\mathrm{LE}_m(s)$ represents the Legendre polynomial of order m. The zero- and first-order polynomials are represented as follows:

$$\mathrm{LE}_0(s) = 1 \quad \text{and} \quad \mathrm{LE}_1(s) = s \tag{2}$$

With a suitable order m, one k dimensional pattern is extended using the expansion block to one t dimensional pattern using Eq. (1), where $t = m * k + 1$. Then, a set of weights with length t is initialized in a random fashion and a weighted sum of the extended input vector is calculated as follows:

$$\mathrm{WS} = \mathrm{wt}_i \mathrm{LE}_i(s) \tag{3}$$

Finally, the weighted sum passes through the learning component usually containing a tanh() or log sigmoid function to produce the output. Then, in the training process, the randomly assigned weights are tuned by a weight updating algorithm reducing the error between network output and actual output.

2.2 Details of Proposed Fraud Detection Model

Most of the time, financial institutions do not disclose the sensitive customer transaction details for the privacy purpose. So as mentioned by several researchers, the unavailability of real-world credit card transactional data has been appeared as one of the biggest issues in performing research to identify credit card fraud transactions. In this study, two data sets collected from UCI repository are taken for experimental verification of the model. After collecting data, it is preprocessed to deal with the missing values. Then, the attribute values are normalized so that all values fall within a small specified range and none of the attribute value overwhelms the other one. Then, a training set containing two-third of the normalized dataset is prepared leaving the remaining three-third in the testing set. After that a suitable order for the expansion block and activation function for the learning component is decided for the LENN. A set of weights is initialized for the network, and the network is trained using a weight updating algorithm. In this study, the weights of LENN are trained using a backpropagation algorithm with a tanh() function used in the learning component as follows:

$$\begin{aligned}
&\mathrm{wt}_{i,k} = \mathrm{wt}_{i,k-1} + \mathrm{lr} \times e_{k-1}(1 - y_{k-1})^2 \times \mathrm{LE}_i(s)\\
&\text{where } \mathrm{wt}_{i,k} = \text{weight at } k\text{th instance}\\
&e_{k-1} = \text{error at } k-1\text{th instance}\\
&y_{k-1} = \text{output at } k-1\text{th instance}\\
&\mathrm{lr} = \text{learning rate}
\end{aligned} \tag{4}$$

After predefined iterations, the training process is stopped and the weights of the network are fixed for testing. By setting appropriate threshold value, the class labels of the input data set are calculated, and the model performance in terms of accuracy is evaluated for training and testing data. The overall LENN-based fraud detection framework is shown in Fig. 1.

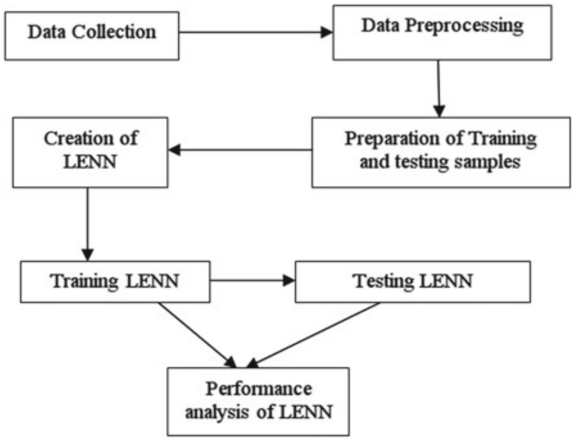


Fig. 1 LENN-based fraud detection framework

3 Experimental Result Discussion

To illustrate the model performance, we have conducted a simulation over two credit card datasets such as German and Australian data set collected from the UCI machine learning repository. Total number of instances originally available for Australian dataset is 690 with total 14 features and for German dataset is 1000 with 20 features. There is a combination of both discrete and continuous values available in both dataset. The collected data are normalized by min-max normalization in the data preprocessing step. Then, 500 instances are kept for training, whereas 190 are kept for testing purpose of Australian dataset. Similarly for German dataset, 800 instances are kept for training and 200 are kept for testing purpose. Next the LENN structure is created with number of input nodes equal to the number of features of training sample of the dataset, one expansion block and one learning component with tanh activation function at the output layer. The network performance has observed with three expansion order, i.e., 3, 4, 5, and four different threshold values used for getting final class level at the output layer. By performing 20 separate simulations, its average performance in terms of classification accuracy is reported in Tables 1 and 2 for both the dataset.

Analyzing the results it is observed that the LENN is providing better result with expansion order 4 and threshold value 0.4 for German dataset and with expansion order 2 and threshold value 0.6 for German dataset. To analyze the impact of learning rate over network performance, further simulation is done with different learning rates keeping the expansion order and threshold value fixed. The corresponding findings are listed in Tables 3 and 4. For German dataset, it is found that higher training and testing accuracy is observed with 0.02 learning rate. In case of Australian dataset though with 0.02 and 0.06 learning rate is giving same result, but higher training accuracy is observed with 0.06 learning rate. Finally, the model is also compared with a CHFLANN, MLP and DT-based fraud detection model, and the corresponding output is presented in Table 5 for both the dataset. Final comparison of the four

Table 1 Output of German credit card data set with different expansion order and threshold values

Threshold	Expansion order	Training accuracy	Testing accuracy
0.4	2	77.33	75.20
	3	77.46	76.80
	4	**78.26**	**77.60**
0.5	2	74.53	73.60
	3	75.33	74.80
	4	75.60	73.06
0.6	2	72.13	70.40
	3	72.53	70.00
	4	73.06	70.40
0.7	2	70.40	70.00
	3	70.66	70.80
	4	70.40	70.40

Table 2 Output of Australian credit card data set with different expansion order and threshold values

Threshold	Order	Training accuracy	Testing accuracy
0.4	2	86.29	84.88
	3	85.90	82.55
	4	85.71	84.88
0.5	2	85.90	86.04
	3	86.10	85.46
	4	85.52	86.04
0.6	**2**	**87.83**	**87.79**
	3	85.90	86.04
	4	86.10	86.04
0.7	2	86.29	86.04
	3	85.90	86.04
	4	87.45	82.55

Table 3 Output of German credit card data set with different learning rates

Threshold	Order	Learning rate	Training accuracy	Testing accuracy
0.4	**4**	**0.02**	**78.26**	**77.60**
		0.04	76.00	73.60
		0.06	73.06	70.00
		0.08	71.20	70.00
		0.10	68.93	62.80
		0.12	59.33	55.20

Table 4 Output of Australian credit card data set with different learning rates

Threshold	Order	Learning rate	Training accuracy	Testing accuracy
0.6	**2**	0.02	86.87	**87.79**
		0.04	87.45	87.20
		0.06	**87.83**	**87.79**
		0.08	87.64	86.62
		0.10	87.83	86.62
		0.12	87.83	86.62

Table 5 Comparative result of LENN with other fraud detection models

Data set	Model	Testing accuracy
German	**LENN**	**77.60**
	CHFLANN	77.00
	MLP	77.50
	DT	66.00
Australian	**LENN**	**87.79**
	CHFLANN	86.32
	MLP	85.26
	DT	85.26

techniques clearly illustrates that LENN outperformed CFLANN, MLP and DT for both dataset.

4 Conclusion

With the unraveling tactics of fraudsters, developing efficient fraud detection model has always been a challenge for the financial industry. This study has explored the performance of a LENN-based classifier for detecting credit card frauds. This study clearly reveals a comparative performance of LENN, CHFLANN, MLP and DT over two different data sets for credit card fraud detection. Results illustrate that LENN gives better detection accuracy than other models. Future research will include meta-heuristic approaches for feature ranking of credit card dataset as well as for the improvement of learning ability of LENN.

References

1. Ngai, E.W., Hu, Y., Wong, Y.H., Chen, Y., Sun, X.: The application of data mining techniques in financial fraud detection: a classification framework and an academic review of literature. Decis. Supp. Syst. **50**(3), 559–569 (2011)
2. West, J., Bhattacharya, M.: Intelligent financial fraud detection: a comprehensive review. Comput. Secur. **57**, 47–66 (2016)
3. Bhattacharyya, S., Jha, S., Tharakunnel, K., Westland, J.C.: Data mining for credit card fraud: a comparative study. Decis. Support Syst. **50**(3), 602–613 (2011)
4. Duman, E., Ozcelik, M.H.: Detecting credit card fraud by genetic algorithm and scatter search. Expert Syst. Appl. **38**(10), 13057–13063 (2011)
5. Sahin, Y., Duman, E.: Detecting credit card fraud by ANN and logistic regression. In: IEEE International Symposium on Innovations in Intelligent Systems and Applications, pp. 315–319 (2011)
6. Sahin, Y., Bulkan, S., Duman, E.: A cost-sensitive decision tree approach for fraud detection. Expert Syst. Appl. **40**(15), 5916–5923 (2013)
7. Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., Baesens, B.: APATE: a novel approach for automated credit card transaction fraud detection using network-based extensions. Decis. Support Syst. **75**, 38–48 (2015)
8. Quah, J.T., Sriganesh, M.: Real-time credit card fraud detection using computational intelligence. Expert Syst. Appl. **35**(4), 1721–1732 (2008)
9. Halvaiee, N.S., Akbari, M.K.: A novel model for credit card fraud detection using Artificial Immune Systems. Appl. Soft Comput. **24**, 40–49 (2014)
10. Behera, T.K., Panigrahi, S.: Credit card fraud detection: a hybrid approach using fuzzy clustering & neural network. In: IEEE Second International Conference on Advances in Computing and Communication Engineering, pp. 494–499 (2015)
11. Zakaryazad, A., Duman, E.: A profit-driven artificial neural network (ANN) with applications to fraud detection and direct marketing. Neurocomputing **175**, 121–131 (2016)
12. Mishra, M.K., Dash, R.: A comparative study of Chebyshev functional link artificial neural network, multi-layer perceptron and decision tree for credit card fraud detection. In: IEEE Conference on Information Technology, pp. 228–233 (2014)
13. Dash, R., Dash, P.K.: Prediction of financial time series data using hybrid evolutionary Legendre neural network: evolutionary LENN. Int. J. Appl. Evol. Comput. **7**(1), 16–32 (2016)
14. Rout, A.K., Dash, P.K., Dash, R., Bisoi, R.: Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach. J. King Saud Univ. Comput. Inform. Sci. **29**(4), 536–552 (2017)
15. Dash, R.: Performance analysis of an evolutionary recurrent Legendre polynomial neural network in application to FOREX prediction. J. King Saud Univ. Comput. Inf. Sci. (2017). https://doi.org/10.1016/j.jksuci.2017.12.010
16. Dash, R., Dash, P.K.: MDHS–LPNN: a hybrid FOREX predictor model using a Legendre polynomial neural network with a modified differential harmony search technique. In: Handbook of Neural Computation 459–486 (2017). https://doi.org/10.1016/b978-0-12-811318-9.00025-9

Time Series Analysis of Rainfall for Puri District of Odisha Using ARIMA Modelling

Subhadra Mishra, Deepak Kumar Swain, Debahuti Mishra, and Gour Hari Santra

Abstract The importance of rainfall for crop is very precious to get the good yield. Timely prediction of rainfall will enlighten the farmers for crop planning. In this research article, time series analysis of rainfall is carried out for Puri District. Suitable ARIMA model is identified on the basis of parameters. The data of *Kharif* rainfall showed a good result with ARIMA (1, 1, 1). The variability in prediction and actual is identified which provides consistency of model. August month received the highest average precipitation of 332.32 mm, whereas December received the lowest precipitation of 5.35 mm. Future crop plan and different agronomic practices for crops can be done accurately after the prediction of rainfall.

Keywords Time series · Rainfall · AIC · ARMA · ARIMA

1 Introduction

Climate change crashes are fluctuating in different parts of the country. The changes in climate patterns had established effects on agriculture. Odisha is one of the poorest

S. Mishra (✉)
Department of Computer Science and Application, CPGS, OUAT, Bhubaneswar, Odisha, India
e-mail: mishra.subhadra@gmail.com

D. K. Swain
Department of Agriculture Statistics, IAS, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: dsagrico14@gmail.com

D. Mishra
Department of Computer Science and Engineering, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: debahutimishra@soauniversity.ac.in

G. H. Santra
Department of Soil Science and Agricultural Chemistry, IAS, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: santragh@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_43

states of India. The major economics of state is agriculture. The biggest challenge for farm community is global climate change effect. Nowadays, regular natural calamities were distressing the farmers to move away from farming. The climate situation triggers an alarming point for new policy and plan for the betterment of farm community. Rainfall is the most important climatic element that influences agriculture [1]. Southwest monsoon is the vital source of water supply for *Kharif* crops which enters in the second half of June and effects continue till October first week. The annual normal rainfall of the state is 1451.2 mm, with a unimodal distribution. From mid-June to September, the state receives more than 80% of the rainfall. The pattern of rainfall is highly unpredictable in amount, distribution and most importantly in time [2]. The total count of rainy days may decrease along with a marginal increase of 7–10% in annual rainfall by the year 2050 for the subcontinent. Climate in Odisha has undergone several changes. Traditional six seasons has been agriculturally decreased to two, i.e., summer and rain. The numbers of rainy days are narrowing from 120 days to 90 days. Based on the rainfall prediction, variety of crop and water resource management can be planned in time. Any deviation will lead to crop failure or else damage level high.

Puri District belongs to coastal belt of Odisha covering an area of 3479 km^2. It belongs to Mahandi River basin. Rivers like Daya, Prachi, Kushabhadra, Devi and Bhargavi are major sources for drainage. The total cultivable land of Puri District is 188,475 ha. Major *kharif* crops are paddy, oilseed, jute and pulses [3]. The mean annual rainfall and rainy days are 1491 mm 57 days, respectively [4].

The present study is focused on time series modelling of rainfall of Puri District for different months (June–December) of *kharif* season. Different parameters based on ARIMA models are evaluated, and their consistency is examined. Box and Jenkins [5] had described these models in his literatures. Some literature of Walter [6] extended the research on it. The ARIMA method is used because the characteristic of considered variable, i.e. rainfall is stationary (has a mean and constant variance also covariance is independent) [7]. This model is used to forecast tourism development for Lithuana [8]. Some researchers applied these models in other sector to get good results [9].

2 Materials and Method

2.1 Data Collection

The time series data of rainfall on monthly basis (June–December) for *Kharif* season from 1983 to 2013 were obtained from Orissa Agricultural Statistics Year Book, 1983–2013, published by Directorate of Agriculture and Food Production, Government of Odisha, Bhubaneswar [10].

2.2 Methodologies

ARIMA Model

An Autoregressive Integrated Moving Average (ARIMA) model is one of the best suitable methodologies for analysing a time series. The speciality of the ARIMA model provides accurate forecasting of historical data of concerned variable.

Box and Jenkins in 1976 has introduced this ARIMA model. This methodology is applicable for stationary time series; i.e. the mean, variance and covariance are constant over time. In non-stationary time series data set, the classical regression results are not valid. In ARIMA,

AR Autoregressive models
MA Moving average models
I The number of lags used in differencing the data.

The mathematical formula of autoregressive (AR) model, moving average (MA) model and ARIMA model are shown in Eqs. (1), (2) and (3), respectively.

Autoregressive (AR) Models

$$Y_t = \emptyset_0 + \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \cdots . + \emptyset_p Y_{t-p} + \varepsilon_t \tag{1}$$

where

$\emptyset_t$ Coefficients to be estimated
p Number of lags.

"p" is a parameter for the model.

Moving Average (MA) Models

$$Y_t = \mu + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \cdots \omega_q \varepsilon_{t-q} \tag{2}$$

where $\boldsymbol{\omega_q}$ = coefficients which are to be estimated and ε_t are the error terms.

"q" represents the number of error terms used in model. It is also a parameter for the model.

ARMA Models

After combining AR and MA models, we will obtain an ARMA (p, q) model which is given below:

$$Y_t = \emptyset_0 + \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \cdots . + \emptyset_p Y_{t-p} + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \cdots \omega_q \varepsilon_{t-q} \tag{3}$$

Differences
Stationarity can be obtained from a non-stationary time series by applying the difference rule. The notation "*d*" indicates the number of times that the time series is differenced to obtain a stationary series.

ARIMA Notation
ARIMA (p, d, q): it represents a p-order autoregressive model with q-order moving average having d times differenced time series.

ARIMA has three different stage modelling approaches—(i) identification, (ii) estimation, (iii) diagnostic checking and forecasting [11].

In identification stage, autocorrelation (ACF) and partial autocorrelation function (PACF) are examined. AR and MA are identified from them.

In estimation stage, models are identified and coefficients are examined. Estimated models are compared with Akaike information criterion (AIC) [12].

In diagnostic stage, the goodness of fit of the model and assumptions of white noise, i.e. autocorrelation, homoscedasticity and normality, were tested. The final stage is to forecast for future, if all the steps carried out firmly.

3 Result and Discussion

The descriptive statistics of the rainfall in the study area is given in Table 1.

All the analyses are carried out in the *R* using forecast package.

The highest average rainfall received during the month of August with 338.32 mm in same month lowest recorded rainfall 175.8 mm and highest is 826 mm. The next best month for receiving average rainfall is July with 295.01 mm, and minimum recorded rainfall is 99.2 mm. In the month of November and December, the average received rainfall is 54.49 mm and 5.35 mm, respectively.

ARIMA (1, 0, 1) model shown in Table 2 tried for fitting monthly annual rainfall for different months of Kharif season, the lowest AIC values recorded in the month of December is 252.64 units where there is negligible amount of rain is received.

Table 1 Descriptive statistics of rainfall in Puri District during different months of south-west monsoon season

Month	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
June	75.5	117.93	159.95	177.54	206.58	493.20
July	99.2	235.20	265	295.01	334.13	609.20
August	175.8	245.50	317.1	338.32	377.93	826.50
September	57.3	176.55	236.35	248.11	272.73	501.90
October	4	62.38	116.3	161.14	192.98	586.40
November	0	6.10	21.75	54.49	77.88	268.10
December	0	0.00	0	5.35	0.48	63.50

Table 2 Evaluation of ARIMA (1, 0, 1) for monthly rainfall with AIC values

ARIMA (1, 0, 1)	Month	AIC value	Predicted value (2013)	Actual value (2013)	% Deviation
	June	361.89	164.41	224.7	26.83
	July	376.04	267.14	329.6	18.95
	August	388.69	375.68	272.5	−37.86
	September	370.12	231.87	206.7	−12.18
	October	386.88	171.22	653.9	73.81
	November	347.28	31.57	0.0	–
	December	252.64	0.00	0.0	–

Table 3 Evaluation of ARIMA (1, 1, 1) for monthly rainfall with AIC values

ARIMA (1, 1, 1)	Month	AIC value	Predicted value (2013)	Actual value (2013)	% Deviation
	June	352.82	171.73	224.7	23.57
	July	371.76	294.01	329.6	10.80
	August	379.45	337.61	272.5	−23.89
	September	361.7	246.16	206.7	−19.09
	October	378.01	160.63	653.9	75.43
	November	339.12	43.68	0.0	–
	December	249.59	5.85	0.0	–

The lowest positive per cent of deviation was recorded in the month of July with 18.95%, and AIC value is 376.04 units; the predicted rainfall during the month was 267.144 mm; and actual rainfall is 329.60 mm. The lowest negative per cent was observed during the month of September with −12.81% the predicted rainfall which is 231.87 mm but the actual rainfall is 206.7 mm.

The ARIMA (1, 1, 1) shown in Table 3 also reveals that the lowest AIC was recorded during December month with 249.59 units. Lowest per cent deviation was recorded during the month of July with 10.80% with actual and predicted rainfalls which are 329.60 mm and 294.01 mm, respectively.

Rainfall Pattern of Puri District The rainfall of each months of the Kharif season of Puri District is shown in Figs. 1, 2, 3, 4, 5, 6 and 7.

4 Conclusion

From the above discussion, prediction of rainfall from ARIMA (1, 1, 1) shows efficient than the ARIMA (1, 0, 1) for different months of Kharif season in the current

Fig. 1 Trend of rainfall for June month

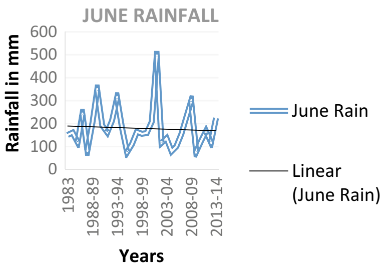


Fig. 2 Trend of rainfall for July month

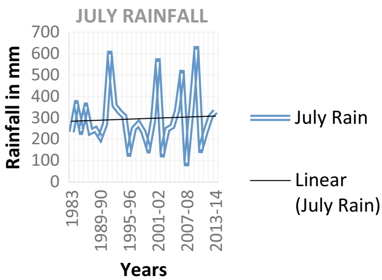


Fig. 3 Trend of rainfall for August month

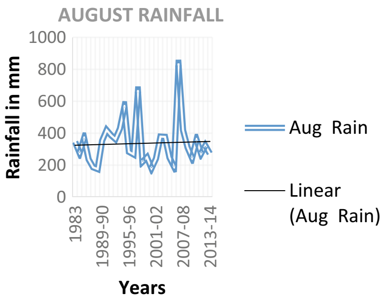
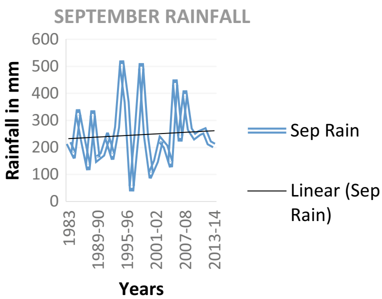


Fig. 4 Trend of rainfall for September month



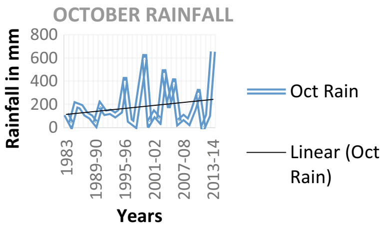


Fig. 5 Trend of rainfall for October month

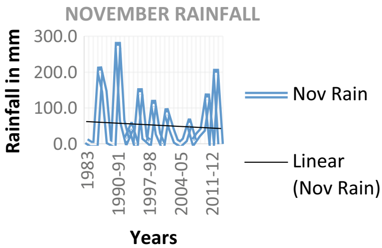


Fig. 6 Trend of rainfall for November month

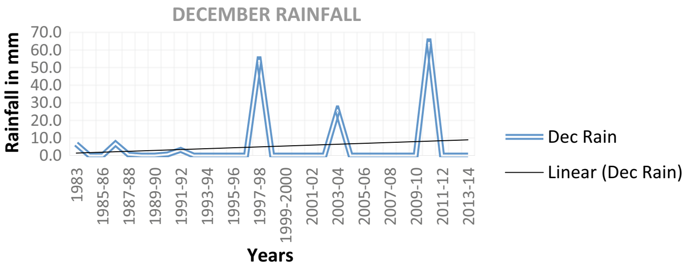


Fig. 7 Trend of rainfall for December month

study area. The AIC values for ARIMA (1, 1, 1) and ARIMA (1, 0, 1) also reveal the same for different months, so use of ARIMA (1, 1, 1) for predicting rainfall is beneficial to carry out any agricultural activities. As rainfall is one of the parameters for the crop production, so the prediction of crop production can also be obtained after analysing the time series of rainfall. Similarly for other district, the time series analysis can be done for rainfall and also for other parameters of yield production.

References

1. Das, S.R.: Rice in Odisha. IRRI Technical Bulletin No. 16. Los Baños (Philippines): International Rice Research Institute (2012)
2. Central Soil and Water Conservation Research and Training Institute, CSWCRTI Vision 2030 (2011)

3. Central Ground Water Board: Ground Water Information Bulletin. Puri District, Odisha (2013)
4. Baliarsingh, A., Nanda, A., Mohapatra, A.K.B.: Effect of rainfall and temperature on rice yield in Puri district of Odisha in India. Pharma Innov. J. **7**(4), 899–903 (2018)
5. Box, G.E.P., Jenkins, G.M.: Time Series Analysis: Forecasting and Control, revised edn., Hoden-Day, San Francisco (1976)
6. Vandaele, W.: Applied Time Series and Box-Jenkins Models, p. 417. Academic Press Inc., Orlando, FL (1983). ISBN: 10: 0127126503
7. Gujarati D.N.: Essential of Econometrics. McGraw-Hill Co., New York (2009)
8. Brinskiene, M., Rudzkiene, V.: Modeling and forecasting of country tourism development in Lithuania. J. Environ. Eng. Landsc. Manag. **13**, 116–120 (2005)
9. Brooks, C.: Introductory Econometrics for Finance. Cambridge University Press, UK (2002)
10. Orissa Agricultural Statistics Year Book, 1983–2013. Published by Directorate of Agriculture and Food Production, Government of Odisha, Bhubaneswar
11. McLeod, A., Lennox, W.: Advances in Box-Jenkins modeling: model construction. Water Resour. Res. **13**, 577–586 (1977)
12. Mcleod, A.I.: Diagnostic checking of periodic auto regression models with application. J. Time Ser. Anal. **15**, 221–233 (1995)

Background Modeling and Elimination Using Differential Block-Based Quantized Histogram (D-BBQH)

Satyabrata Maity, Nimisha Ghosh, Krishanu Maity, and Sayantan Saha

Abstract The proposed work uses differential block-based quantized histogram ($DBBQH$) feature extraction technique to background modeling and elimination for foreground extraction. $D - BBQH$ uses a statistical approach to estimate the temporal consistency of pixel intensity concerning location. Besides, the block-based technique is less noise prone than the same of pixel-based processing and includes local features to estimate the background more efficiently than that of global processing. The proposed approach used a low-dimensional feature set, which requires less amount of time for background modeling or foreground extraction. A benchmark $SBMI$ dataset, which incorporates complex video data, is used to complete the experimentation of the proposed approach, and the results prove the effectiveness of the proposed approach.

Keywords Background modeling · Quantized histogram · Block-based processing · Temporal consistency

1 Introduction

The use of video information is increasing exponentially with the rapid change in technological advancement in the domain. The amount of space required to store a video is much higher than any other form of information, till people love to watch the

S. Maity (✉) · N. Ghosh · K. Maity · S. Saha
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: satyabratamaity@soa.ac.in

N. Ghosh
e-mail: nimishaghosh@soa.ac.in

K. Maity
e-mail: krishanumaity@soa.ac.in

S. Saha
e-mail: sayantansaha@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_44

video for experiencing the fact visually. The main reason behind the volume of the video is redundant data to enrich its quality in terms of human perseverance. The huge amount of redundancy makes the process of actual information extraction manually from the video extremely difficult. This phenomenon inspires many researchers to design automatic video analysis systems.

There are two main kinds of redundancies present in the video-based information—(A) spatial redundancy: the repetition of the similar information spatially by increasing the resolution of a frame to make it more visually prominent to the spectators and (B) temporal redundancy: the replication of similar frames more to increase the smoothness in the transition of visual facts. Most of the time, redundancy is incorporated to increase the visual quality of a video. On the other hand, redundancy is negatively proportional to the information entropy. Hence, to extract the actual information, it is required to eliminate the redundancies; i.e., redundancy increases the processing overhead. A huge part of the redundancy in a video is due to the background information, which remains static throughout the scene. On the other hand, the foreground is the dynamic part with significant information entropy. Hence, it is the most required areas for analysis to get through the vital information of the video. Thus, background elimination for the extraction of the foreground is a crucial step of video analysis, which is essential for various applications of video processing like human action detection, video surveillance and monitoring, medical image analysis, sports, entertainment, robotic vision, etc. The background and foreground segmentations help us in two ways: Specifically, (A) the removal of redundant parts results in the required area to analyze the video and (B) reduce the computational complexity for analyzing the video.

In this paper, we have proposed a background modeling technique based on block-based probabilistic feature extraction technique, which is adaptive to change in background information. There are several environmental conditions in outdoor and indoor scenes in the real-world scenarios. Some of the common challenges to the background modeling problem are bootstrapping, illumination changing, ghosting effect, dynamic background, foliage, missing or abundant objects in the scene, etc. The dataset, we used to verify our results, is containing all these difficulties.

The human eyes can easily sense the background measuring its rigidity and are easily distinguishable from the continuous changing foreground. The main motivation is to estimate the most statistically rigid information to model the background, and use an algorithm to eliminate it from the current frame to extract the foreground. Technically, a series of intensities are used to represent an image and the number of intensity levels is proportional to the clarity of the image/video. The human eye cannot sense all the intensity levels. Neighboring pixels with close intensity values add negligible information. On the other hand, sometimes these negligible details become the cause of false alarm. Suppose, in a video scene, where a tree with moving leaves is in the background, if we try to extract the foreground, the moving leaves will be treated as foreground. Thus, pixel-level processing for background modeling is affected by minute details. Hence, the use of a quantizing bin is used to reduce the number of levels. When a collection of close intensity levels are treated as a single bin, the bin is called quantized bin. Let there be L intensity levels, and those are divided into Q number of bins. If $L = Q \times K$, then K is the size of each bin. This approach does not only minimizes the complexity but also helps to restrict the false alarm. The quantized bin is extracted from each of the blocks in a frame. Every frame is divided into smaller size blocks to emphasize local processing. The histogram of this quantized bin is used as the feature of each block termed as block-based quantized bin ($BBQH$). The proposed approach is trying to estimate the rigid part of a sequence of consecutive frames. Hence, it considers the unmoved features among every two consecutive frames. Thus, the feature of the proposed approach is computed by comparing each of the pairs of consecutive frames, which is called differential block-based quantized histogram ($D - BBQH$). This phenomenon reduces the effect of noise due to minute details, and the block-level comparison reduces the effect of intermittent object movement but giving importance to the local values. The changing profile stores the values between every two consecutive frame pairs; a sudden change in the profile refers to the huge change in the same; if this change in a longer period the proposed method re-estimates the background and thus makes the procedure adaptive.

2 Related Work

A huge volume of work has been done in this domain, and the interested rereads are redirected to [1, 2] for completing their review in this field. The adaptive mixture model proposed by Staffur and Grimson [3] is the pioneering work in this field. The work is based on occurrence statistics of intensity values in particular pixel position in k consecutive frames. The mixture model uses the Gaussian formula to determine the rigid and moving value of a particular location. Many other methods based on pixel-based processing applied formula like averaging [3], median [4], fuzzy-based selection, etc. Some works used frame-based global processing to estimate the rigidity like global histogram [5], optical flow-based technique [5], wavelet transform, etc. The kind of procedures extracts frame-based features to estimate the rigidity. The background is modeled by comparing those features among k consecutive frames.

Pixel-based processing is mostly affected by the noise as it considers every pixel individually. On the other hand, frame-level processing would not consider the local disparity as it focuses on the global behavior of the frames. Hence, the block-level processing is proposed that combines the advantages of both of the other strategies and reduces the shortcomings of the same. The concept is to employ the block-based feature instead of global and pixel-based processing to take care of local changes as well as to reduce the effect of noise. The key contributions, in this research work, can be summarized as follows:

- Proposal of a block-based statistical background modeling technique is based on block-based temporal features.

- The newly introduced $D - BBQH$ feature vector eliminates the active parts to estimate the background. Active parts are the distinguishable part between any consecutive frame pairs.
- The proposed method can handle several irregularities in the video data like ghosting effect, noisy conditions, bootstrapping, camouflaged conditions, etc.
- The complexity of comparing block-based features instead of pixel-based comparison is much lesser as the size of the feature vector is much lesser than the number of pixels. On the other hand, counting sort like strategy helps to estimate the D-BBQH features in a single pass.

3 Proposed Work

The key operations of the proposed approach are described in Fig. 1 that include preprocessing, D-BBQH feature extraction and background modeling, background elimination and foreground extraction based on the $D - BBQH$ feature.

3.1 Preprocessing

This step is used to prepare a frame for feature extraction in a much better way as the contrast normalization removes small illumination changes and the smoothing operation reduces the intra-regional tiny details.
Contrast normalization helps to extract information from low contrast image and reduce the noise due to the variation of illuminations. Contrast normalization needs two distinct values, viz. upper limit (ul) and the lower limit (ll) of the range to which the intensity values are to be normalized. The formula provided in Eq. (2) is used for normalization, where hv and lv are the highest intensity and lowest intensity of the present image.

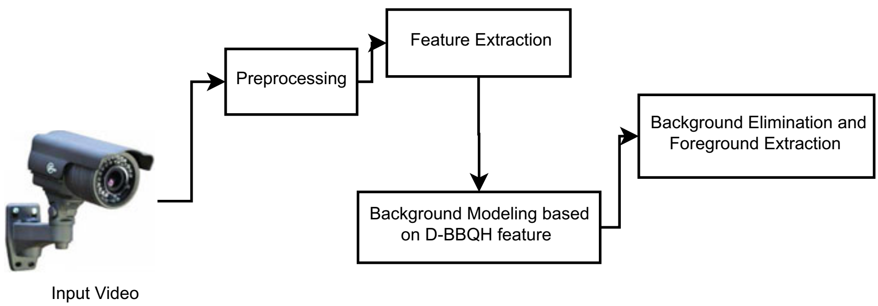


Fig. 1 Flow diagram of the proposed approach

$$P_{\text{out}} = (P_{\text{in}} - \text{ll}) \left(\frac{\text{ul} - \text{ll}}{\text{hv} - \text{lv}} \right) + \text{ll} \tag{1}$$

A simple 3×3 fast averaging filter is used for smoothing operation for any image to reduce the intra-regional disparity; for a color image, we apply the same operation in three channels.

3.2 D-BBQH Feature Extraction and Background Modeling

3.2.1 Elimination of Inconsistent Pixels

This step eliminates the inconsistent pixels among each pair of frames. The consistency of any pixel is defined in terms of consecutive occurrence of similar intensity at the same pixel position. The similarity is checked concerning a prescribed or adaptive threshold. The differential frame includes the pixel if the absolute difference is below a threshold; otherwise, we make it '-1', which means the pixel will not be considered for feature extraction. Thus, the value -1 indicates the pixel is inconsistent. We formulate the concept of temporal consistency in terms of consistency in intensity values among the consecutive frames of any pixel of the frame instead of only the occurrence probability of the same. $\Delta f(x, y, t)$ is the difference between the pixel of location (x, y) of two consecutive frames computed using Eq. (2). Equation (3) is used to compute $\Omega(x, y, t)$, the resultant value for future processing. If $\delta f(x, y, t)$ is less than a certain threshold THchg, then the value of the current frame is taken for consideration; else the value is set to be -1, as the change is greater than background consideration value. β will help to estimate the feature vector. The response computed using δ will be stored in β for further use (Fig. 2).

$$\Delta f(x, y, t) = |f(x, y, t) - f(x, y, t + 1)| \tag{2}$$

$$\Omega(x, y, t) = \Big\{ f(x, y, t), if(\Delta(x, y, t) \geq TH_{\text{chg}} - 1, \text{Otherwise} \tag{3}$$

$$\beta_{ij}(q, 2) = p_q \times \delta^2 \tag{4}$$

3.2.2 Consistency and Occurrence Probability

The proposed approach checks the temporal consistency of any information using two parameters, specifically occurrence probability of the information (ρ_i is computed using Eq. (5), and the consistency of the information (σ_i) is computed using Eq. (6). The occurrence probability ρ_i is the probability of any information in a particular area in t consecutive frames. On the other hand , σ_i is defined in terms of the occurrence

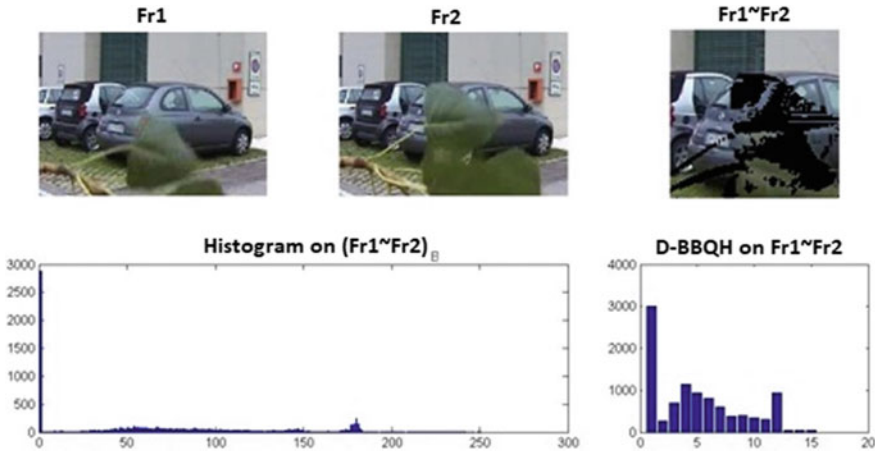


Fig. 2 Pictorial representation of consistency; clockwise from the left top corner Frame 1, Frame 2, removal of inconsistent pixel to get the resultant frame (Fr1 Fr2), D-BBQH of the resultant frame, histogram of the resultant frame

in successive frames. The weighted response of temporal consistency W_i is then computed using Eq. (7).

$$\rho_i = F_i/N \tag{5}$$

$$\sigma_i = \sum(\mathrm{Per}_{ik}), \quad \text{where } \mathrm{Per}_{ik} = n_k(n_k+1)/2 \tag{6}$$

$$W_i = (\rho_i \times \sigma_i) \tag{7}$$

Let a list of information in 24 consecutive frames be given in $LIST = [1, 1, 1, 1, 2, 3, 2, 6, 2, 1, 1, 1, 2, 4, 2, 5, 2, 3, 2, 2, 4, 3, 2, 5]$. All F_i, ρ_i, σ_i and W_i are shown in the table below. The weighted response W_i of 1 is 4.4667, which is the highest among the list, though the occurrence probability of 2 is the highest. Hence, we can conclude that the response value of any information depends on not only its occurrence probability, but also the consistency value. This property helps a lot to reduce several unwanted errors, which occur more frequently but do not persist for long time.

$D-BBQH$ feature is extracted on the resultant frame of the previous operation using Eqs. (8) and (9) .

$$\beta_{ij} = (\sum \gamma_q)/\beta_{ij}(q, 2), \text{ where each} \gamma_q \epsilon q \text{th bin of} \beta_{ij} \tag{8}$$

$$\beta_{ij}(q, 2) = p_q + \delta^2 \tag{9}$$

3.2.3 Feature Analysis for Background Modeling

$D - BBQH$ features of k consecutive frames are used to estimate the backgrounds. We compute the weight of each bin using Eq. (10), which includes consistency and occurrence probability of each bin. A threshold is used to select the $D - BBQH$ features of the background frame depending on the weighted response. This feature vector of background frame β_{BAC} is used for foreground extraction.

3.3 Background Elimination and Foreground Extraction

It is mainly needed to extract the foreground of the current frame by eliminating the background information. Comparison is done block-wise, and if the difference between frequencies of a bin is greater than a certain threshold, the corresponding position of the foreground feature vector will store 1.

$$\beta\text{FRG}_{ij} = \begin{cases} 1 \text{ if } \beta_{ijt}(q, 2) - \beta\text{BAC}(q, 3) > THi_2 \neq j_1 \\ 0 \text{ Otherwise.} \end{cases} \tag{10}$$

3.4 Foreground Extraction

The first foreground feature vector of a block is used to estimate the foreground area. The bin location (Bin) of every pixel is first computed, where F is the current frame and B is the number of bins. If the value of the foreground feature vector is 1 for the bin of the corresponding block, then that pixel is considered as a foreground pixel.

4 Results and Analysis

The proposed approach is verified on SBMI [?] dataset, and a comparative study with several renounced methods of related literature justifies the proficiency of the proposed approach.

4.1 Implementation Assumptions

$D - BBQH$ features are extracted from each of the frames of a video. There is a storage, which can store $D - BBQH$ features of N number of frames, and it discards the $D - BBQH$ features of the first frame when the D-BBQH features of $(N + 1)$th frame are needed to insert, discards the second frame when $D - BBQH$

features of $(N+2)$th frame are needed to insert and so on. Background modeling is done by merging the features of a K number of D-frames. The value of K is set to the frame rate of any video if there is no movement in the initial frames, and if movement found, it is set to twice of the same. There is no extra cost needed for feature extraction when background updating is done as the feature of the current frame is extracted for foreground extraction. There are two other thresholds, namely: Frame differentiation threshold is set to the twice of the standard deviation of first k frames, and the background bin selection threshold is set to the highest weight of the block divided by $\sqrt{2}$.

4.2 Dataset

We use Scene Background Modeling and Initialization (SBMI) [6] dataset to describe the efficiency of our algorithm. The dataset includes 14 image sequences and their ground truth backgrounds with different levels of difficulties like foliage, abandoned and lost objects, late foreground movement, foreground covers certain areas most of the part of the scene, etc. This dataset is adopted from the SBMI2015 workshop, and this is a very recent dataset as the last update was done in January 16. The challenges of the videos are given in Table 1.

4.3 Evaluation Parameters [7]

The Web site provides the script for evaluating results concerning six metrics, which are used in the literature for background estimation. GT is denoting the ground truth image, and CB is the estimated background of any background modeling approach. The six metrics are used to estimate the difference between GT and CB images and evaluate the effectiveness of the difference.

Table 1 Frequency, consistency, occurrence probability and the weighted response of the corresponding sample

Sample	F_i	σ_i	ρ_i	$W_i = \sigma_i \times \rho_i$
1	7	16	0.297	4.667
2	9	10	0.375	3.75
3	3	3	0.125	0.375
4	2	2	0.083	0.1667
5	2	2	0.083	0.1667
6	1	1	0.0417	0.0417

- Average Gray-level Error (AGE): The absolute difference between gray-level GT and CB images. This is a global estimation of differences, and the smaller value is more appreciable.
- Percentage of Error Pixels (pEPs): This parameter confirms the percentage of the number of wrongly estimated background pixels by using a reasonable threshold value (20 is suggested by the link). The lower percentage of error proves the better quality of an approach.
- Peak Signal-to-Noise-Ratio (PSNR): The standard formula is used to extract the PSNR of the computed background frame compared to the ground truth of the same. The value provides the superiority of the information over the noise. Thus, the bigger value is more desirable.
- Multi-scale Structural Similarity Index (MS-SSIM): This parameter is proposed by is used to estimate the perceived visual distortion by using structural distortion, and the greater value determines the lesser distortion.

The evaluation of the results of the proposed approach with the other related research is based on the above four previously described parameters.

The proposed method used a $D - BBQH$ feature for background estimation, so the output of our method includes quantized bins instead of pixels. So when we map the features into a frame, it selects the highest bin which matches the temporal values of a location. This is needed to test our background modeling results with the same of the ground truth values. On the other hand, if we check the same concerning feature values, the evaluation procedure will not support it. Though, our method produces the best results in the case of AGE, MS-SSIM and PSNR values, which are used for global error estimation, multi-scale structural similarity, and peak signal to noise ratio respectively. The results in Table 2 include the average values of seven videos for several parameters of the proposed and fourteen other methods of the related research.

The proposed approach produced efficient background modeling results instead of several difficulties of several videos enlisted in Table 2. Change in lightning conduction, background noise, jitters and the presence of shadow/camouflage condition may lead to false detection. The first foreground movement is important for checking

Table 2 Comparative results on six evaluation parameters of the proposed approach and the same of the related research work

Approaches	Average AGE	Average pEPs	Average MS-SSIM	Average PSNR
SC-SOBS [8]	6.32	8.02	0.93	29.89
WS2006 [9]	6.50	4.69	0.934	27.605
Photomontage [10]	7.42	6.58	0.905	30.9901
CA2008 [11]	13.676	10.04	0.91	25.17
GROUSE [12]	19.166	0.359	0.915	24.37
Proposed	4.72	2.79	0.954	30.057

the robustness against bootstrapping. If there is no foreground movement, we do not need to store or process those frames. Finally, if we need to update the background, then two consecutive modeled background frames of any scene assist to track the abandoned or missing object as the abandoned object(s) are the new inclusion and missing object(s) are the eliminated part of the two modeled frames. The SBI dataset includes all of this kind of difficulties, and our proposed approach produces effective results in spite of all the difficulties.

5 Conclusion

In this paper, we have shown a novel block-based hybrid background modeling technique in terms of frame differentiating and block-based feature extraction. The feature extraction technique is newly driven depending upon the consistency measurement of the pixel values of any block among successive frames. The results of the proposed methodology over the $SBMI$ dataset and the comparative results on the related research work show the efficiency of the same. This methodology focused on the consistency of static information by removing the active part of any consecutive frame pairs. Till the method does not provide a hundred percent accurate results, so the scope of further improvement is possible.

References

1. Goyette, N., Jodoin, P.M., Porikli, F., Konrad, J., Ishwar, P.: Changedetection. net: a new change detection benchmark dataset. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE, pp. 1–8 (2012)
2. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In Proceedings of 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), vol. 2, pp. 246–252 (1999)
3. Ekinci, M., Gedikli, E.: Background estimation based people detection and tracking for video surveillance. In: Yazc, A., Sener, C (eds.) International Symposium on Computer and Information Sciences, vol. 2869 (2003)
4. McFarlane, N.J.B., Schofieldi, C.P.: Machine vis. apps. In: Yazc, A., Ener C (eds.) Computer and Information Sciences - ISCIS 2003 (1995)
5. Mohamad Hoseyn Sigari: Naser Mozayani, and Hamid Reza Pourreza. Concepts and application, Fuzzy running average and fuzzy background subtraction (2008)
6. Maddalena, L., Petrosino, A.: Towards benchmarking scene background initialization. In: Murino, V., Puppo, E., Sona, D., Cristani M., Sansone C (eds.) New Trends in Image Analysis and Processing—ICIAP Workshops. (2015)
7. Maddalena, L., Bouwmans, T., Petrosino, A.: Scene background initialization: a taxonomy. Pattern Recog. Lett. **96**, 3–11 (2017)
8. Wang, H., Suter, D.: A novel robust statistical method for background initialization and visual surveillance. In Nayar S.K., Shum HY (eds.) Computer Vision ACCV 2006. ACCV 2006. Lecture Notes in Computer Science, vol. 3851 (2006)
9. Bouwmans, T.: Recent advanced statistical background modeling for foreground detection—a systematic survey. Recent Pat. Comput. Sci. **4**, 147 (2011)

10. Maddalena, L., Petrosino, A.: The sobs algorithm: what are the limits? In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 21–26 (2012)
11. Chen, C., Aggarwal, J.K.: An adaptive background model initialization algorithm with objects moving at different depths. In: 2008 15th IEEE International Conference on Image Processing, pp. 2664–2667 (2008)
12. Balzano, L., Wright, S.J.: On grouse and incremental svd. In: 2013 5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), pp. 1–4 (2013)

Spectral Clustering Based Fuzzy *C*-Means Algorithm for Prediction of Membrane Cholesterol from ATP-Binding Cassette Transporters

Ramamani Tripathy, Rudra Kalyan Nayak, V. Saravanan, Debahuti Mishra, Gayatri Parasa, Kaberi Das, and Priti Das

Abstract In the human genome, biological membrane plays an imperative part in all cell organisms. Generally, cell membrane consists of two components such as lipids and proteins. Phospholipids and cholesterols are both treated as most abundant lipids in plasma membrane where cholesterol molecules are mostly hydrophobic in nature. In this paper, we described interaction of membrane cholesterol with transmembrane proteins. Cholesterol is a major constituent in membrane proteins which is not uniformly distributed in biological membrane, and it has other responsibility like membrane fluidity and lipid raft. In most eukaryotes, ATP-binding cassette (ABC) transporters are represented as superfamily among all transmembrane proteins. Here we focus on target of ABC transporters by membrane cholesterol and counting down of the binding sites between them. Basically, membrane cholesterol binds the

R. Tripathy (✉)
Department of Master of Computer Application, United School of Business Management, Bhubaneswar, Odisha, India
e-mail: ramatripathy1978@gmail.com

R. K. Nayak · V. Saravanan · G. Parasa
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation (Deemed to be University), Guntur, Andhra Pradesh, India
e-mail: rudrakalyannayak@gmail.com

V. Saravanan
e-mail: reachvsaravanan@gmail.com

G. Parasa
e-mail: gayathriparasa20@gmail.com

D. Mishra · K. Das
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: debahutimishra@soa.ac.in

K. Das
e-mail: kaberidas@soa.ac.in

P. Das
Department of Pharmacology, SCB Medical, Cuttack, Odisha, India
e-mail: pritidaspatnaik@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_45

membrane proteins for predicting valid signature motif from ABC transporter which gives significant value. In this paper, a computational approach has been implemented which is based on spectral clustering with fuzzy *C*-means algorithm to find different types of amino acid sequences from the binding region between ABC transporter and membrane cholesterol. Finally from our experiment, we achieved better prediction accuracy results which have biological significance.

Keywords FCM · Spectral clustering · CRAC · CARC · ATP-binding cassette transporters

1 Introduction

Cholesterol is treated as the foremost component of plasma membrane in human genome. Cholesterol is a waxy component, and it regulates membrane fluidity and also generates an effective diffusion barrier between intracellular to extracellular. Cholesterol modulates the function of several membrane proteins so it is also recognized as fundamental constituent of plasma membrane. The consensus motif of cholesterol is seen either in forward pattern or backward pattern according to their N-C-terminus. Using CRAC and CARC methods, we predict cholesterol valid signature motifs from ABC transporter. Cholesterol plays an important role in hormone production, bile secretion, and vitamin D production. Due to the excess cholesterol in human body, it imposes various diseases such as coronary heart disease, stroke, type-2 diabetes, high blood pressure. Human ABC transporter acts as a superior superfamily of transmembrane protein which consists of 48 well-characterized genes. Basically, 48 ABC genes can again be separated into seven subfamilies according to their phylogenetic analysis which is ranging from ABCA to ABCG. Individual subfamilies are responsible for different diseases. Mutations in several human ABC transporter genes have been linked to several genetic disorders such as neurological disease, retinal degeneration, cholesterol and bile transport defects, cystic fibrosis, drug response, and anemia. ABC transporters take part in important roles in case of metabolism and lipid transport in plasma membrane [1–8].

To date, structural analysis and protein target prediction have been very tedious task for current research. For solving these types of problems, most of the researchers are continuing their study using various data mining techniques such as fuzzy, SVM, neural network, and naïve Bayesian to predict membrane proteins [9, 10]. Therefore, our work is concentrating on a hybrid model that is spectral clustering-based fuzzy *C*-means approach which is very appropriate with our dataset and gives higher prediction accuracy of cholesterol motif from human ABC transporter. The organization of the paper is shown below: In Sect. 2, data set and proposed model are presented. In Sect. 3, the projected spectral clustering-based fuzzy *C*-means methodology is elucidated, and in Sect. 4, the experimental evaluation is represented, and finally in Sect. 5, conclusion part is depicted.

2 Data Set with Proposed Model Description

2.1 Description of Dataset

All membrane protein sequences were retrieved from UniProt database [11]. As protein sequences are arranged in a helical manner so all are retrieved in their respective way in text format like helical 1 to helical 17. The imperative purpose of current work is to recognize target amino acid motif of cholesterol from transmembrane sites of ABC genes. Membrane cholesterol only targets on either *N*-terminus or *C*-terminus in forward/backward manner so many combinations may arise in CRAC and CARC method.

For the prediction of valid cholesterol signature from ABC genes, we are using both forward (CRAC) (L/V–$X_{(1-5)}$Y–$X_{(1-5)}$–R/K) and backward (CARC) (R/K–$X_{(1-5)}$–Y–$X_{(1-5)}$–L/V) patterns. We target our transmembrane sequences with cholesterol sequences using our proposed model.

2.2 Proposed Model

Figure 1 summarizes the schematic outline of the proposed model. In this model, first we retrieved all helical files of ABC protein which varied from H1 to H17. Then we are constructing a cholesterol dictionary using CRAC and CARC formula according to sliding window concept. Then we found target motif of cholesterol from ABC proteins, and then we applied spectral clustering-based fuzzy *C*-means algorithm for

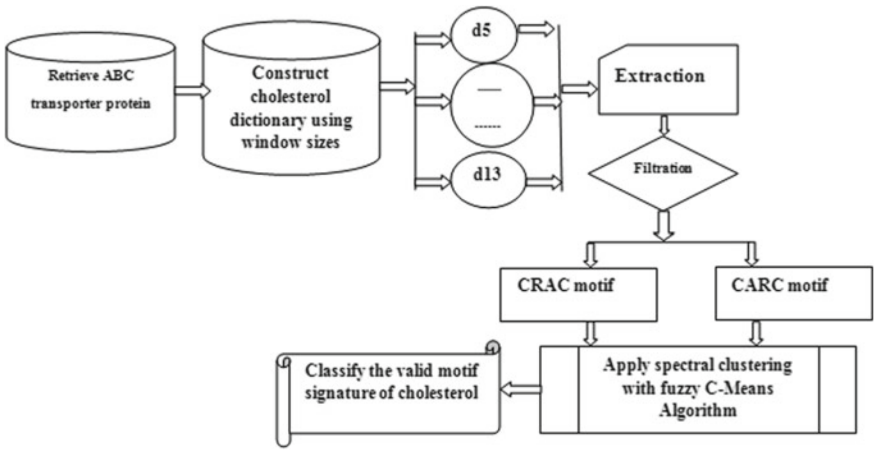


Fig. 1 Schematic layout of our proposed model for valid signature motif of cholesterol

Table 1 Each possible motif-type combination of cholesterol and X represents combination of amino acid that can vary from (1 to 20)

Type of Motif	Forward direction (L/V–$X_{(1–5)}$–Y–$X_{(1–5)}$–R/K)	Backward direction (R/K–$X_{(1–5)}$–Y/F–$X_{(1–5)}$–L/V)
11, 12, 13, 14, 15	L/V1Y1R/K, L/V1Y2R/K, …, L/V1Y5R/K	K/R1Y/F1L/V, K/R1Y/F2L/V, …, K/R1Y/F5L/V
21, 22, 23, 24, 25	L/V2Y1R/K, L/V2Y2R/K, …, L/V2Y5R/K	K/R2Y/F1L/V, K/R2Y/F2L/V, …, K/R2Y/F5L/V
31, 32, 33, 34, 35	L/V3Y1R/K, L/V3YR/K, …, L/V3Y5R/K	K/R3Y/F1L/V, K/R3Y/F2L/V, …, K/R3Y/F5L/V
41, 42, 43, 44, 45	L/V4Y1R/K, L/V4YR/K, …, L/V4Y5R/K	K/R4Y/F1L/V, K/R4Y/F2L/V, …, K/R4Y/F5L/V
51, 52, 53, 54, 55	L/V5Y1R/K, L/V5Y2R/K, …, L/V5Y5R/K	K/R5Y/F1L/V, K/R5Y/F2L/V, …, K/R5Y/F5L/V

accurate clustering of motif among all data points. Here we find out valid signature motif of cholesterol which has pathological significance.

All cholesterol dictionaries are constructed using CRAC and CARC formula. Both forward and backward motifs are described with their window size in Table 1. For example, considering window-sized 8 containing length $L = 8$ we can have motif types MT = {14, 23, 32, 41} and signature motif of cholesterol lies in the form of L/VX1YX4K/R, L/VX2YX3K/R, L/VX3YX2K/R, L/VX4YX1K/R and K/RX1Y/FX4L/V, K/RX2Y/FX3L/V, K/RX3Y/FX2L/V, and K/RX4Y/F X1L/V for both forward and backward methods, respectively. Wherever, X position can vary from one motif sequence to other motif residue. From table, it is clear that according to the position of Y/F, membrane cholesterol signature motif is changed.

The purpose of our paper is to find valid cholesterol motif sequences using both CARC and CRAC from ABC family. After filtration, a novel spectral with fuzzy C-means method is required for finding the most significant motif from the huge number of obtainable motif sequences. Here we are using spectral clustering with FCM which assigns each data point after all individual motif sequences are found which have clinical relevance for drug discovery.

3 Methodology Description

3.1 Spectral Clustering

From last few decades, clustering approaches are broadly used in a variety of fields for pattern recognition and also for other aspects. In this technique, clusters are formed on the basis of similarity and dissimilarity of data. The data are grouping in such a manner that similar data are in one cluster and dissimilar data are forming another

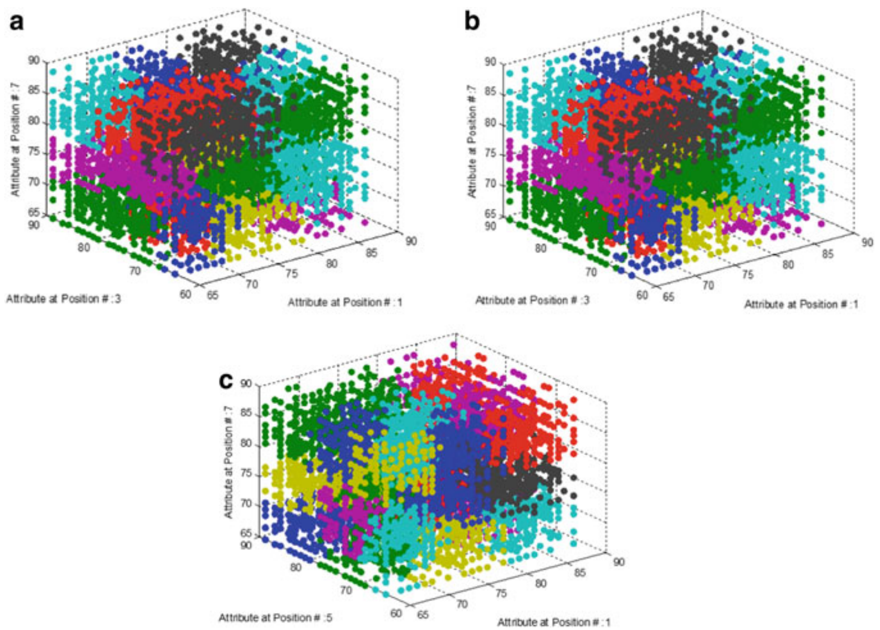


Fig. 2 Plot diagram of different cluster points of Y according with their position shown in (**a**) (**b**) (**c**)

clusters. Based on this concept, all clustering algorithms work. In case of spectral clustering, each data points are considered as graph's nodes. Graph portioning concept is used here. Generally, three important steps must be taken for computing the cluster point such as calculate the similarity matrix, project the data onto a low-dimensional space, finally construct the cluster center [12–14]. In this paper, we have applied spectral clustering with FCM tool for predicting cholesterol signature.

Algorithm: Spectral Clustering

Input:

Data set $X = \{x_1, \ldots, x_n\}$, Initailize σ *scaling*

begin:

Step 1: Using scaling scheme data are Preprocessing

Step 2: Build an Affinity matrix $A_{jk} = \exp\left(-\frac{||d_j - d_k||^2}{2\sigma^2}\right)$

Step 3: Build a Laplacian matrix $L^{norm} := I - P^{-\frac{1}{2}} A P^{-\frac{1}{2}}$

Step 4: Calculate the k largest Eigen vectors $u_i, \ldots, u_k$ of L

Step 5: matrix is $U=[u_i, \ldots, u_k] \in R^{n \times k}$

Step 6:Form a matrix Z from U as $Z_{ij} = \frac{u_{ij}}{(\sum_j u_{ij}^2)^{1/2}}$

Step 7: Cluster each Z using k-means

Step 8: Assign the U_i to cluster $j$iffZ_i is assign to cluster j

3.2 Fuzzy C-means (FCM)

Nowadays, people dealing with huge amount of data in every sector so clustering techniques help people to obtain solution from that fuzzy problem. Therefore, fuzzy *C*-means clustering approach is considered as one of the foremost theories with the growth of technology and broadly used in engineering, scientific disciplines, medicine imaging, pattern detection, and bioinformatics. This algorithm has significant role in case of unsupervised technique for the investigation of data and edifice of models. Basically, FCM utilizes fuzzy partitioning such that a data point can belong to everyone groups with dissimilar membership grades between 0 and 1 [15–18].

Algorithm : Fuzzy C-Means

1. Choose randomly the cluster center;
2. Initialize W=[$membership_{kj}$] matrix, W(0)
   On the basis of the equation (3) $membership_{kj}$ can be calculated
3. Compute the center vectors $center^x = center_j$ with $W^{(x)}$ using equation (4) at x-step
4. Update $W^{(x)}$, $W^{(x+1)}$ using equation (3);
   If $\| W^{(x+1)} - W^{(x)} \| < \varepsilon$ or the smallest amount of Cluster is accomplished, then STOP; else revisit step 2.

4 Experimental Evaluations

From Tables 2 and 3, it is observed that valid significant cholesterol signatures motifs are discovered using spectral with fuzzy *C*-means on basis of Protein*ID*, *Gene name*, *Helix*, Conserve red *motif signature*, *Start/End* where cholesterol motif matched with ABC protein, and *Motif type*. The outcomes obtained from CRAC and CARC where clearly illustrate about all combinations of motif. From our study, it is revealed that backward motif has more targeted region in comparison with forward motif.

5 Conclusion

In our current work, we have elaborated about the effect of membrane cholesterol on the membrane proteins. Membrane cholesterol is a vital component of eukaryotic membranes and plays an important role in membrane organization, dynamics, and function and have modulator role on membrane proteins like G-protein coupled receptors, ABC transporter. Our experiment tells that cholesterol motif targets the protein sequence of ABC transporter which is the combination of amino acids in *N*–*C*-terminus. However, the present work reveals an extra and unforeseen mode of cholesterol target site. From this, we found that backward motif of cholesterol has much more target binding site than forward motif. It was experimented that both forward/backward motifs have greater influence in the transmembrane helices

Table 2 Forward signature motifs for ABC derived from L/V–$X_{(1–5)}$–Y–$X_{(1–5)}$–R/K

Unique protein Id	Gene name of protein	Helix number	Obtained sequence	Motif type
ABCA2	Q9BZC7	h6	VPYMYVAIR	V15R, V33R, L45R, L53R, L54R
ABCA5	Q8WWZ7	h15	LLQYYEKK	L12K, L13K, L21K, L22K, L23K, L31K, L32K
ABCA6	Q8N139	h7	LLLALYFDK	L22K, L32K, L42K, V2SK, L35K, V45K
ABCAS	O94911	h7	LALAIYFEK	L22K, L42K, V13K, L23K, L33K
ABCA9	Q8IUA7	h7	LVLTLYFDK	L22K, V32K, L42K.
ABCA12	Q86UK0	h6	VENELSYVLK	L12K, V52K, V15K, V13K, L33K, L35K, V53K, V55K
ABCB2 (TAP1)	Q03518	h7	LSLFXWYLVR	L12R, L32R, L52R, L21R, L31R, V41R
ABCB10	Q9NRK6	h3	VIYGRYLRK	V11R, V14R, V15K, V41R, V42K
ABCC1	P33527	hl7	LQVTTYLNWLVR	V25R, L45R
ABCC3	O15438	h1	LPCYLLYLR	L11R, L24R, V44R, L51R, V14R, L54R, L34R
ABCC7(CFTR)	P13569	h5	LSVLPYALIK	L13K, V23K, L43K
ABCC8	Q09428	h5	VIRVRRYIFFK	V23K, V53K, V14K, L44K
ABCC10	Q5T3U5	h1	VLSACYLGTPR	L34R, V44R, L24K, V31K, V44K, L51K
ABCD3	P28288	h2	VNNFLKYGLNELK	L15K, V55K, V14R, V24R, L54R
ABCG1	P45844	h6	LRLIAYFVLRYK	L25K, L43R, L45K, L23R

(continued)

Table 2 (continued)

Unique protein Id	Gene name of protein	Helix number	Obtained sequence	Motif type
ABCG4	Q9H172	h6	LRLLAYLVLRYR	L13R, L12K, L15R, V22K, L25R, L45R, L23R, L,43R, L32K

Table 3 Backward signature motifs for ABC derived from K/R–$X_{(1-5)}$–Y/F–$X_{(1-5)}$–L/V

Unique protein Id	Gene name of protein	Helix name	Obtained sequence	Motif type
ABCA1	O95477	h9	RKGFFAQIVL	R34L; R33V; R25L; R24V; K24L; K23V; K14V; K15L; K55V; K52L; K51V
ABCA1	O95477	h13	KIPSTAYVVLTSV	
ABCA13	Q86UQ4	h10	RMYWFTNFL	R33L, R15L
ABCB1	P08183	h12	KLMSFEDVLLV	K35V, K34L, K33L, K32V
ABCC1	P33527	h12	KAIGLFISFL	
ABCC2	Q92887	h2	KQVFVGFLLIL	K53L, K51L, K24L, K23L, K14V, K54L, K53L, K52V, K23L, K22L, K21V
ABCC2	Q92887	h6	KALFKTFYMVLL	
ABCD2	Q9UBJ2	h2	KKPRTFIIKL	K43L, K33L, R13L
ABCD4	O14678	h2	KDLEGFKTLTFL	K45L, K42L, K34L, K32V
ABCG1	P45844	h6	KLYLDFIVL	K42L, K41V, R41L, R32L, R31V, K14V, K15L
ABCG2	Q9UNQ0	h4	KPKADAFFV	K51V, K45L, K31V
ABCG4	Q9H172	h6	KLYMDFLVL	K42L, K41V, R32L, R31V, K13L, K14V, K15L
ABCG5	Q9H222	h4	RFGYFSAALL	R34L, R33L, R25L, R24L

regions of those that were modulated by membrane cholesterol or involved in cholesterol transport. As a result our method, spectral with fuzzy C-means works well on these datasets and reported number of CARC signature motif sequences which have biological significance on human diseases in ABC superfamily.

References

1. Baier, C., Fantini, J., Barrantes, F.: Disclosure of cholesterol recognition motifs in transmembrane domains of the human nicotinic acetylcholine receptor. Sci Rep. **1**(69), 1–7 (2011). https://doi.org/10.1038/srep00069
2. Pucadyil, T.J., Chattopadhyay, A.: Role of cholesterol in the function and organization of G-protein coupled receptors. Prog. Lipid Res. **45**(4), 295–333 (2006). https://doi.org/10.1016/j.plipres.2006.02.002
3. Epand, R.M., Thomas, A., Brasseur, R., Epand, R.F.: Cholesterol interaction with proteins that partition into membrane domains. An overview. In: Harris, J. (eds.) Cholesterol Binding and Cholesterol Transport Proteins: Subcellular Biochemistry, vol. 51, pp. 253–278. Springer, Dordrecht (2010). https://doi.org/10.1007/978-90-481-8622-8_9
4. Oram, J.F.: Molecular basis of cholesterol homeostasis: lessons from Tangier disease and ABCA1. Trends Mol. Med. **8**(4), 168–173 (2002). https://doi.org/10.1016/s1471-4914(02)02289-X
5. Higgins, F.C.: ABC transporter: from microorganism to man. Annu. Rev. Cell Biol. **8**(1), 67–113 (1992). https://doi.org/10.1146/annurev.cb.08.110192.000435
6. Paila, Y.D., Chattopadhyay, A.: Membrane cholesterol in the function and organization of G-protein coupled receptors. In: Harris J. (eds) Cholesterol Binding and Cholesterol Transport Proteins: Sub cellular Biochemistry, vol. 51, pp. 439–466. Springer, Dordrecht (2010). https://doi.org/10.1007/978-90-481-8622-8_16
7. Gurevich, V.V., Gurevich, E.V.: Molecular mechanisms of GPCR signaling: a structural perspective. Int. J. Mol. Sci. **18**(12) (2017). https://doi.org/10.3390/ijms18122519
8. Tripathy, R., Mishra, R., Konkimalla, V.K., Nayak, R.K.: A computational approach for mining cholesterol and their potential target against GPCR seven helices based on spectral clustering and fuzzy C-means algorithms. J. Intell. Fuzzy Syst. Preprint, 1–10 (2018). https://doi.org/10.3233/jifs-169589
9. Karchin, R., Karplus, K., Haussler, D.: Classifying G-protein coupled receptors with support vector machines. Bioinformatics **18**(1), 147–159 (2002). https://doi.org/10.1093/bioinformatics/18.1.147
10. Andrew, A.M.: An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, vol. 18(6), pp. 687–689. Cambridge University Press (2000). https://doi.org/10.1017/S0263574700232827
11. http://www.uniprot.org
12. Chen, A.C., Lerman, G.: Spectral clustering based on local linear approximations. Electr. J. Stat. **5**, 1537–1587 (2011). https://doi.org/10.1214/11-EJS651
13. Zare, H., Shooshtari, P., Gupta, A., Brinkman, R.R.: Data reduction for spectral clustering to analyze high throughput flow cytometry data. BMC Bioinform. **11** (2010). https://doi.org/10.1186/1471-2105-11-403
14. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: Advances in Neural Information Processing Systems (2002)
15. Fogarassy, A.V., Feil, B., Abonyi, J.: Minimal spanning tree based fuzzy clustering. Trans. Enformat. Syst. Sci. Eng. **8**, 7–12 (2005)
16. Krishnapuram, R., Keller, J.M.: A possibilistic approach to clustering. IEEE Trans. Fuzzy Syst. **1**(2), 98–110 (1993). https://doi.org/10.1109/91.227387

17. Suganya, R., Shanthi, R.: Fuzzy *C*-means algorithm—a review. Int. J. Sci. Res. Publ. **2**(11) (2012)
18. Chitralegha, M., Thangavel, K.: Protein sequence motif patterns using adaptive fuzzy *C*-means granular computing model. In: International Conference on Pattern Recognition, Informatics and MobileEngineering. IEEE (2013). https://doi.org/10.1109/icprime.2013.6496454

Exploration of Hyperparameter in Extreme Learning Machine for Brain MRI Datasets

Ashwini Kumar Pradhan, Kaberi Das, Debahuti Mishra, and Subhasis Mishra

Abstract In past decades, it has been observed that learning speed of feed-forward neural network was too slower than its required speed, which was a major drawback for their applications. The reason behind this problem is that all the parameters of system are tuned iteratively. This paper proposes a modern learning algorithm called extreme learning machine (ELM) in which parameters are not tuned. ELM is one of the best performing classifiers. Its architecture consists of various interconnected layers made up of several neurons. With the help of subjective weights, the input layers and hidden layers are interconnected and provide output with the help of activation function or transfer function. Activation functions are nonlinear, and various activation functions may give distinct output on same dataset. Only one out of every activation function is appropriate for each sort order issue. Results demonstrate that the accuracy of ELM is predominantly influenced by the number of neurons for a certain dataset. This paper demonstrates the fluctuation of normal test precision with different activation functions using brain MRI datasets.

Keywords ELM · Single-hidden-layer feed-forward neural networks · Activation functions · Classification

A. K. Pradhan (✉) · K. Das · D. Mishra
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: ashwini.apply@gmail.com

K. Das
e-mail: kaberidas@soa.ac.in

D. Mishra
e-mail: debahutimishra@soa.ac.in

S. Mishra
Department of Computer Science and Engineering, College of Engineering and Technology, Bhubaneswar, Odisha, India
e-mail: subhasismishra.mishra@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_46

1 Introduction

In prior years, various investigations have been carried out on the classification issues and various solutions have been proposed over it. Out of some standard systems like Bayesian decision theory, multilayer perceptron (MLP) [1, 2], support vector machine (SVM), artificial neural network (ANN), ANN is one of the certified methods which uses back-propagation technique [3] and trial and error techniques where weight is adjusted. It encounters issues like local minima [4], overfitting [5] and more preparing time [6]. SVM is a well-known classifier which depends on measurable hypothesis and gives superior speculation accuracy. SVM experiences issues like high calculation and data dependent.

The word extreme learning machine (ELM) model was given by its fundamental designer Huang [7]. As indicated by ELM designer, the model can deliver high-level accuracy as it takes less time to learn the system as compared to back-propagation. ELM has one hidden layer feed-forward system where parameters among input to hidden layer are taken out randomly and parameters between hidden layer and output layers taken logically by utilizing a numerical idea that is Moore–Penrose (MP) generalized inverse technique. ELM is broadly utilized in different fields [8] like face location, multi-class classification, pattern recognition, image processing, signal processing and so on. We have selected ELM due to its higher learning speed, less learning epochs, less preparing time for parameters, reduced overfitting problem, easy halting criteria, etc. Here, the input weights are generated arbitrarily while an activation function is utilized at hidden layer. Execution of different activation functions similar to sigmoid, triangular bias [9], sign, hard limiter and radial bias [10] gives different results on same dataset. The main objective of this paper is to find the best activation function which gives better accuracy with respect to different numbers of hidden neurons.

This paper is explained in four sections. Section 1 describes the introductory part. Section 2 deals with related work. In Sect. 3, the working principles of ELM have been elaborated. Section 3 includes concise idea about activation functions of ELM. Experimental performance of ELM on various datasets and result analysis are discussed in Sect. 4 which is followed by conclusion.

2 Related Works

In recent years, researchers have moved toward machine learning approach for classification problems. Huang et al. [7] in 2006 conducted experiment based on some benchmark activation functions in ELM for classification problem and found that ELM learns thousands of times faster than traditional learning algorithms. Duch et al. [9] in 1999 used functions such as circular, conical, Gaussian function to conduct a systematic comparison between them and found that the fair comparison of transfer functions would be very difficult. Chen [11] in 2016 primarily introduced a technique on NeuroEvolution of Augmenting Topologies using combinatorial activation functions (arctan and sigmoid) to enhance expressive power of ELM.

3 Extreme Learning Machine (ELM)

ELM has single-hidden-layer feed-forward neural system (SLNF) [2, 12] which provides creative answers for classification problems. We divide the original dataset into two random datasets for adjustment of classifiers named as training set and other testing set.

Figure 1 shows the schematic diagram of ELM in which x and y are input and output vectors correspondingly and g is an activation function. The parameters w (input weights) and b (hidden layer biases) are randomly generated in ELM, and the β (output weights) can be mathematically derived through the inverse operation. ELM not only is faster but also avoids local minima. The output equation of ELM is as follows:

$$O_j = \sum_{i=1}^{L} \beta_i g(w_i \cdot x_j + b_i), \quad j = 1, 2, \ldots m \tag{1}$$

Here, O_j represents output vector of Jth input vector, where $i = 1, 2, \ldots, L$, and L represents hidden node count. $\beta = [\beta_1 + \beta_2 + \cdots + \beta_i]^T$ is the output weight matrix.

Given a training set $N = \{(X_i, t_i)| \in R_d,\ i = 1, 2, ..n\}$ and g is activation function. Learning issue can be solved by (2).

$$H * \beta = Y \tag{2}$$

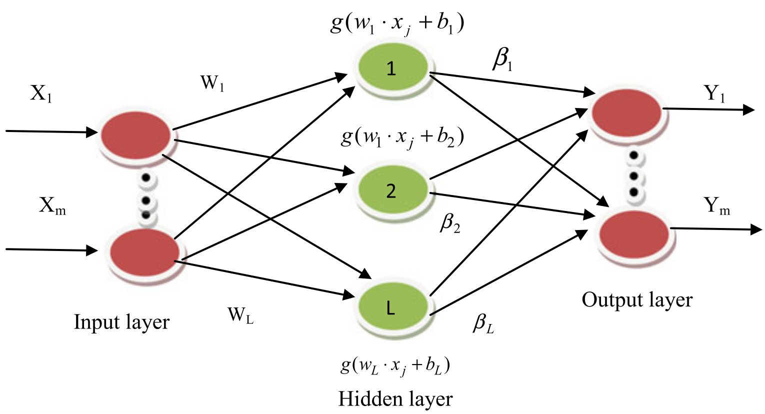


Fig. 1 Schematic representation of ELM

where $Y = [t_1, t_2, \ldots, t_n]^T$ is target value matrix. $H = [hT(X_1), \ldots, hT(X_n)]^T$. β can be evaluated from (3).

$$\beta = H^+ Y \tag{3}$$

where H^+ represents Moore–Penrose generalized inverse of H.

4 Activation Functions

For calculation of output response of neurons, activation functions are used upon weighted sum of inputs. One activation function is used in one layer throughout the network. It is up two types: One is linear, and other is nonlinear activation function. Mainly, nonlinear activation functions are used for classification issues. Some nonlinear activation functions implemented in this paper are as follows.

(i) **Sine Function**: It takes a real value along with returning another real value while ranging in between 1 and -1. Sin library function is used in MATLAB to execute it.

(ii) **Sigmoid Function**: It is an 'S'-shaped activation function which is in the form $F(x) = 1/1 + \exp(-x)$ that range in between 0 and 1. The major drawback of this function is as follows:

- Evaporation of gradient issue.
- Lack of zero-centered output.

(iii) **Radial Bias Function**: It is a distance-based function from origin. This function is utilized through inbuilt library function available inside MATLAB called radbas.

(iv) **Triangular Bias Function**: Triangular inclination limit can work as neural exchange. This limit determines a layer's yield from its known data. The triangular tendency limit is executed with tribas inbuilt function available within MATLAB.

(v) **Hard Limit Function**: It is a threshold-based limiting function that determines the neuron position as 0 and 1. When it touches the threshold, it gives 1 else 0.

5 Experimental Evaluation

Here, the ordinary execution of the five different activation functions is inspected based on accuracy of classification as well as time taken by them. We implement recreation of ELM in MATLAB (R2014a) on Intel CORE i3 system. The output depends on two factors, i.e., number of activation function and number of hidden nodes.

Table 1 Dataset used for classification

Datasets used	Category of datasets	Training instances	Testing instances	Total number of instances
Alzheimer	Binary	60	30	90
Hemorrhage	Binary	140	60	200

5.1 Dataset Depiction

For classification, the entire dataset must be valid and complete. Table 1 represents the selected datasets (Alzheimer dataset and hemorrhage dataset) from Kaggle repository. We divided it into two parts before classifying: One is training and another for testing. Entities were selected randomly to create imbalance datasets for training and testing. The table given below shows the details of dataset in terms of number of records, training set, testing set and category (binary).

5.2 Analysis of Results

For calculation of the performance in ELM, the code executed 12 epochs through equal number of hidden nodes in addition to activation functions on a particular dataset. After implementation, we got randomization within consequence; hence, the average value has been taken for Alzheimer and hemorrhage datasets, which is shown in Tables 2 and 3. Both factors (number of hidden nodes and activation functions)

Table 2 Average classification accuracy and average processing time for Alzheimer dataset

Functions	No. of hidden nodes				
	10	20	30	40	50
Average accuracy					
sin	0.4900	0.5150	0.5050	0.4890	0.4880
Sig	0.6420	0.7530	0.7930	0.8030	0.8190
tribas	0.6150	0.7270	0.7950	0.8070	0.8010
radbas	0.4700	0.4700	0.4700	0.4700	0.4700
hardlim	0.4680	0.4700	0.4730	0.4710	0.4670
Average timing					
sin	0.0025	0.0025	0.0031	0.0025	0.0025
Sig	0.0031	0.0047	0.0025	0.0031	0.0025
tribas	0.0025	0.0025	0.0031	0.0025	0.0025
radbas	0.0047	0.0047	0.0025	0.0025	0.0031
hardlim	0.0049	0.0048	0.0048	0.0064	0.0068

Table 3 Average classification accuracy and average processing time for hemorrhage dataset

Functions	No. of hidden nodes				
	10	20	30	40	50
Average accuracy					
sin	0.5170	0.4850	0.4930	0.4910	0.5060
Sig	0.6120	0.7320	0.7940	0.7680	0.8130
tribas	0.6030	0.7000	0.7580	0.7620	0.7560
radbas	0.3400	0.3400	0.3390	0.3400	0.3400
hardlim	0.3990	0.4030	0.3980	0.4010	0.4030
Average timing					
sin	0.0032	0.0032	0.0036	0.0036	0.0036
Sig	0.0032	0.0036	0.0035	0.0035	0.0031
tribas	0.0041	0.0038	0.0038	0.0038	0.0035
radbas	0.0049	0.0049	0.0049	0.0051	0.0035
hardlim	0.0052	0.0052	0.0055	0.0055	0.0052

play an important role in classification. The number of nodes in the hidden layer is 10, 20, 30, 40 and 50 that has been taken for this experiment.

From Tables 2 and 3, it has been observed that maximum accuracy is obtained by sig activation function with 50 numbers of hidden nodes in both the datasets. The highest average accuracy achieved in both the datasets is 81.30%. Here, the time factor of ELM has not been taken into consideration because ELM is itself much faster than others.

Figures 2 and 3 show the performance graph which represents the summary of experiments on Alzheimer and hemorrhage datasets. The average accuracy is measured on final outcome due to arbitrary accuracy proportion and appears on same dataset with equal number of activation function as well as equal number of hidden

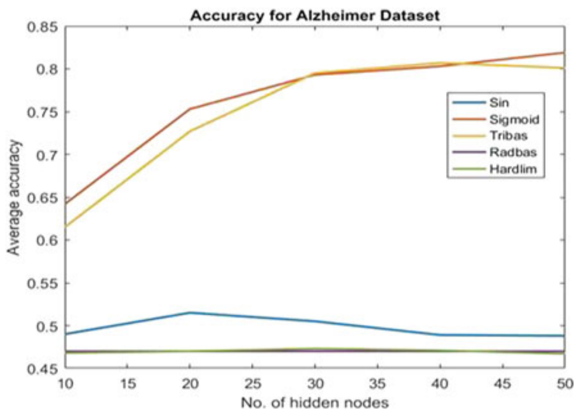


Fig. 2 Training performance of ELM with respect to different activation functions and number of hidden nodes used in Alzheimer dataset

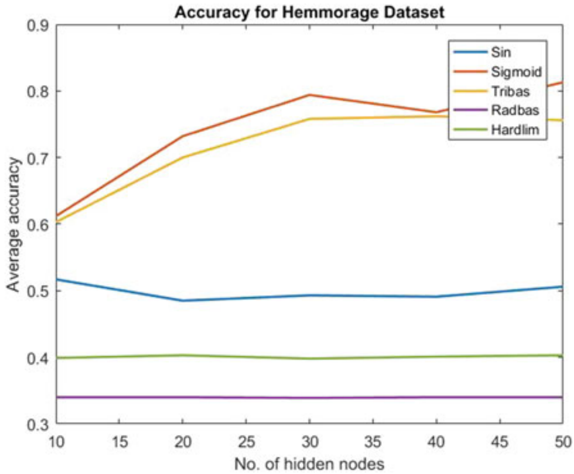


Fig. 3 Training performance of ELM with respect to different activation functions and number of hidden nodes used in hemorrhage dataset

nodes. The following graphs (Figs. 4 and 5) correspond to the accuracy variation on 12 times for different implementations of ELM algorithm using both the datasets. Finally, the sigmoid (sig) activation function gives better accuracy by applying both the datasets.

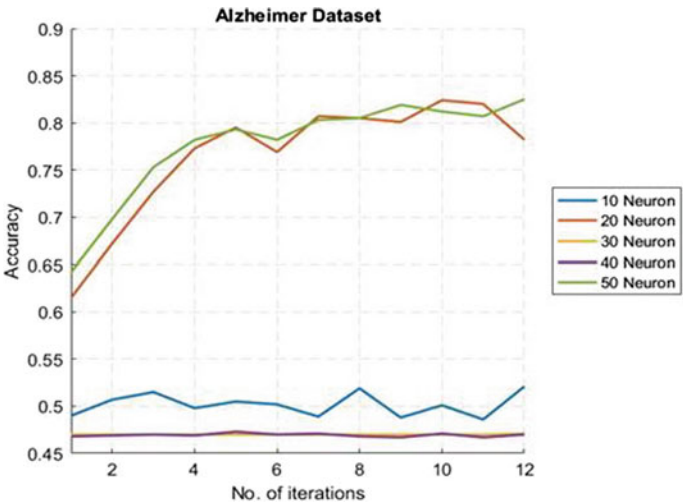


Fig. 4 Graph between accuracy and the 12 iterations used in Alzheimer dataset

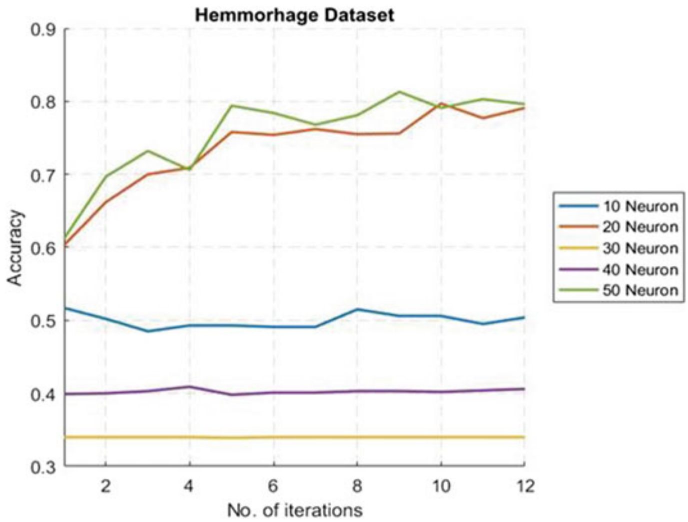


Fig. 5 Graph between accuracy and the 12 iterations used in hemorrhage dataset

6 Conclusion

The correlation of activation functions is performed on both the datasets, and a number of hidden nodes are used in hidden layer. It has been found that by changing the quantity of neurons in hidden layer, the accuracy is also changed and sigmoid activation function achieved a superior execution on 50 numbers of hidden neurons on both Alzheimer and hemorrhage datasets. If we increases the number of hidden neurons that does not insures the accuracy of classifiers. While contrasting handling time, 50 numbers of shrouded neurons additionally assume the best job as contrast with others. In this experimental study, every activation function does not provide equal result intended for given datasets with same number of hidden nodes. Accuracy varies because the additional factors similar to input weight, input size and bias also affect the performance and they are randomly generated at each simulation. This work may be extended in future days. ELM can achieve much better, if instead of using random number generating functions, some techniques for calculation of input weights may be added. This might increase the classification accuracy.

References

1. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley (2012)
2. Kay, S.M.: Detection Theory. Prentice-Hall (1998)
3. Samarasinghe, S.: Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition. Auerbach Publications (2016)

4. Tamura, S., Tateishi, M.: Capability and storage of four layered feedward neural network. IEEE Trans. Neural Netw. (1997)
5. Wang, G., Li, P.: Dynamic Adaboost ensemble extreme learning machine. In: IEEE 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), vol. 3, pp. 53–54 (2010)
6. Yam, J.Y., Chow, T.W.: Feedforward networks training speed enhancement by optimal initialization of the synaptic coefficients. IEEE Trans. Neural Netw. **12**(2), 430–434 (2001)
7. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. Neurocomputing **70**(1–3), 489–501 (2006)
8. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: a new learning scheme of feedforward neural networks. Neural Netw. **2**, 985–990 (2004)
9. Duch, W., Jankowski, N.: Survey of neural transfer functions. Neural Comput. Surv. **2**(1), 163–212 (1999)
10. Simon, H.: Neural Networks and Learning Machines. Simon Haykin, USA (1999)
11. Chen, J.: Combinatorially Generated Piecewise Activation Functions. arXiv Preprint arXiv:1605.05216 (2016)
12. Pandey, P., Singh, R.G.: Analysis of randomized performance of bias parameters and activation function of extreme learning machine. Int. J. Comput. Appl. **135**(1), 23–28 (2016)

Jaya Optimized Extreme Learning Machine for Breast Cancer Data Classification

Santos Kumar Baliarsingh, Chinmayee Dora, and Swati Vipsita

Abstract As the latest World Health Organization (WHO) statistics show, approximately 15% of all cancer deaths among women, i.e., 627,000 in number in the year 2018 are from breast cancer in the USA. To make a definitive cancer diagnosis, a biopsy is a medical practice for maximum types of cancers, as it provides the most accurate analysis of tissue. To avoid the huge number of undesirable breast biopsies, new diagnosis methods based on microarray data analysis have been proposed in the last decades. The microarray analysis can help the physicians to decide whether to carry out a breast biopsy or not. Gene expression profiling based on microarray data has been evolved as an effective procedure for the classification of cancer along with its diagnosis and treatment. Machine learning techniques like artificial neural network have shown significant potential in cancer classification and clinical diagnosis. In this paper, a classification method based on Jaya optimized extreme learning machine (JELM) has been applied on breast cancer microarray data after relevant genes are selected employing Wilcoxon rank sum test. Jaya is used to pre-train the ELM by selecting the optimal input weights and hidden biases. ELM rectifies the difficulties raised by iterative learning techniques such as local minima, improper learning rate, and overfitting. Finally, a comparative result analysis is presented on the achieved classification accuracy by four well-known classifiers available in the literature, namely SVM, *k*NN, NB, and C4.5. Accuracy is used as a performance metric to analyze the efficiency of the classifiers. From the obtained results, it is

S. K. Baliarsingh (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: santoskbaliarsingh@soa.ac.in

C. Dora
Department of Electronics and Telecommunication Engineering, IIIT Bhubaneswar, Bhubaneswar, Odisha, India
e-mail: chinmayee.dora@gmail.com

S. Vipsita
Department of Computer Science and Engineering, IIIT Bhubaneswar, Bhubaneswar, Odisha, India
e-mail: swati@iiit-bh.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_47

observed that JELM classifier achieves better classification accuracy as compared to other schemes.

Keywords Microarray · Extreme learning machine · Wilcoxon rank sum test · Jaya

1 Introduction

In case of women, the three most common cancers are breast, lung, and colorectal [1]. In breast cancer, breast cells are abnormally divided and grown in an unbounded fashion. Worldwide, breast cancer prolongs to be a compelling public health issue. For an effective treatment of breast cancer, recent computer-aided diagnosis techniques based on DNA microarray have exhibited tremendous impact for the identification of cancer-causing genes [2, 3]. However, in the analysis of microarray data, the primary hindrance that remains is the curse of dimensionality issue [4]. By this issue, the computational cost is greatly increased, reducing the classification accuracy. Consequently, the identification of appropriate genes becomes difficult for the accurate analysis of breast cancer microarray data [5]. Many researchers and practitioners have proposed numerous techniques for selection and classification of relevant genes based on machine learning [6–10].

This paper proposes a method in which Wilcoxon rank sum test is used to select relevant breast cancer genes and JELM is employed for the classification. Two primary steps of DNA microarray data analysis are feature selection followed by classification. First, feature selection step has to be carried out since classification step depends on the outcome of the first step. The primary aim of gene selection process is to determine a relevant set of genes that help the researchers to discriminate between different classes of cancers [11]. During the process of selection of genes, sometimes, low-ranked genes are selected where as high-ranked genes are left out [12]. To avoid these problems, an efficient gene selection technique has to be chosen that would enhance classifier's predictive accuracy. All the above criteria can be satisfied when Wilcoxon rank sum test is employed as feature selection technique, and JELM is employed as a classifier.

A brief overview of Wilcoxon rank sum test [13, 14], Jaya optimization algorithm [15], and ELM [16, 17] is described in Sect. 2. An experimental study is given in Sect. 3. Section 4 presents the conclusion.

2 Proposed Method

This section describes the proposed gene selection method followed by the process of classification. Section 2.1 describes Wilcoxon rank sum test as feature selection method. In Sect. 2.2, extreme learning machines are briefly described as a classifier. Section 2.3 presents the proposed Jaya-based ELM.

2.1 Gene Selection Employing Wilcoxon Rank Sum Test

As the feature size of the microarray dataset is very high, the computational cost increases. To reduce the cost and to retrieve the relevant genes from data, a feature selection technique, namely Wilcoxon rank sum test, is applied. The mathematical notation of this test is shown in Eq. 1:

$$m(G) = \sum_{a \in J_0} \sum_{b \in J_1} D\left(\left(E_b^{(G)} - E_a^{(G)}\right) \leq 0\right) \tag{1}$$

where $m(G)$ denotes the magnitude of the difference among the classes. A particular gene is more important to the classifier when this difference is near to 0 or near to the maximum value of $j_0 j_1$ (here $j_0 = |J_0|$, $j_1 = |J_1|$). J_0 and J_1 represent the indices of various classes of samples. D represents the discrimination function. When the value of the expression inside the parenthesis is true, then the result of D is 1, or else it is 0. $E_a^{(G)}$ represents sample a's expression value for the gene G. Now, using Eq. 2, the degree of a gene can be computed as follows:

$$r(G) = \max(m(G),\ j_0 j_1 - m(G)) \tag{2}$$

At last, a ranking of genes is done by their $r(G)$ value and as per the ranking, highest ranked p genes are selected from the input data to form a new subset.

2.2 Extreme Learning Machines

A feed-forward neural network (FNN) is a biologically inspired classification algorithm having one input layer, and one or many hidden layers followed by an output layer. The feature vector is passed between input layers, processed in between, and then the output layer transfers the result to the external environment.

The connections between the layers of the network are associated with weights. These weights allow the propagation of the information against the input layer up to the output layer. When these weights are optimized, the model gets trained. ELM is one such type of FNN with only one hidden layer. Let us assume an FNN having one hidden layer with K hidden nodes. Let $g(x)$ be the activation function to train N different samples represented as $(x_i,\ t_i)$, where $x_i = [x_{i1}, x_{i2}, \ldots x_{in}] \epsilon R^n$ and $t_i = [t_{i1}, t_{i2}, \ldots t_{im}] \epsilon R^m$. The weights of input vectors and biases of hidden layers are taken at random; then, there exist β_i, w_i, and b_i such that

$$f_K(x_j) = \sum_{i=1}^{K} \beta_i G(w_i, b_i, x_j) = t_j, \quad j = 1, \ldots, N \tag{3}$$

where β_i is the weight between the ith hidden and output layers. Outcome of the ith hidden node is denoted as $G(w_i, b_i, x)$ corresponding to the input x. $G(w_i, b_i, x)$ is given by:

$$G(w_i, b_i, x) = g(w_i \cdot x + b_i), \quad b_i \in R^n \tag{4}$$

where w_i represents the weight vector between the input layer and ith hidden layer, and b_i represents the bias of ith hidden layer. The dot product of vectors w_i and x in R^n is denoted by $w_i \ldots x$.

Now Eq. 3 can be represented as

$$H\beta = T \tag{5}$$

where $H(w_1, \ldots, w_K, b_1, \ldots, b_K, x_1, \ldots, x_K)$

$$= \begin{bmatrix} G(w_1, b_1, x_1) & \cdots & G(w_H, b_H, x_N) \\ \vdots & \ddots & \vdots \\ G(w_1, b_1, x_N) & \cdots & G(w_H, b_H, x_N) \end{bmatrix}_{N \times N} \tag{6}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \cdot \\ \cdot \\ \cdot \\ \beta_K^T \end{bmatrix}_{KXm} \quad \text{and} \quad T = \begin{bmatrix} t_1^T \\ \cdot \\ \cdot \\ \cdot \\ t_N^T \end{bmatrix}_{NXm} \tag{7}$$

H represents output matrix of the hidden layer. The output vector of the ith hidden layer is represented by the ith column of H, and the hidden layer output vector for input x_j is shown by the j_{th} row of H.

Always, K (number of hidden nodes) should be less than N (number of samples). Therefore, the training error (ε) will not be zero. The parameters of the hidden layer, w_i and b_i of SHLFNs, are not optimized during training. These values are randomly assigned; hence from Eq. 5, the output layer's weights are computed as:

$$\hat{\beta} = H^{\dagger} T \tag{8}$$

where $H^{\dagger}$ is the Moore–Penrose generalized inverse of H. In this paper, the sigmoidal function is used as the activation function.

2.3 Jaya Optimization Algorithm

Rao et al. [15] proposed a simple and powerful global optimization algorithm, i.e., Jaya for solving any optimization problems that may be constrained or unconstrained.

This is a simple and novel optimization technique from an application point of view. It is an algorithm without parameters, where convergence of the optimal solution is obtained with comparatively lesser number of function evaluations. For the above-mentioned advantages, Jaya algorithm could be used in solving different application-oriented optimization problems. The key concept of the algorithm is that the solution for a given problem should be obtained with avoiding the worst solution and moving toward the best solution.

Considering a problem where P is the number of candidate solutions (i.e., population size, $i = 1, 2, \ldots P$) and for each candidate solution, there are d number of decision variables (i.e., $j = 1,2, \ldots, d$). At any iteration z, X^z_{best} is denoted as the best solution obtained among all the candidate solutions and X^z_{worst} the worst solution. During zth iteration, if the value of the jth decision variable for the ith candidate solution is $X^z_{i,j}$, then $X^z_{i,j}$ is updated as per Eq. 9:

$$X^z_{i,j} = X^z_{i,j} + \mu^z_{1,j}\left(X^z_{j,\text{best}} - \left|X^z_{i,j}\right|\right) - \mu^z_{2,j}\left(X^z_{j,\text{worst}} - \left|X^z_{i,j}\right|\right) \tag{9}$$

where the best and worst values of decision variable j are $X^z_{j,\text{best}}$ and $X^z_{j,\text{worst}}$, respectively. $X^z_{i,j}$ is the updated value of $x^z_{i,j}$. During zth iteration, $\mu^z_{1,j}$ and $\mu^z_{2,j}$ are the two random numbers in the range [0, 1]. If $X^z_{i,j}$ yields a better valued objective function, it is accepted in $x^z_{i,j}$. Thus, at the end of each iteration, modified $x^z_{i,j}$ is again allowed to participate in the next iteration. This procedure repeats till the end of number of iterations.

2.4 Proposed Jaya-Based ELM

The generalization characteristic of the ELM primarily relies on two network parameters: (i) the input weights and (ii) the hidden bias. The random initialization of these two parameters causes two serious issues: (a) More number of hidden neurons are required by the ELM than the conventional gradient-based learning, and therefore, the performance of the ELM begins to be slow, and (b) ELM creates ill-conditioned hidden layer output matrix which causes little generalization performance.

Therefore, to resolve the problem, a recently developed global optimization algorithm known as Jaya optimization is used which adaptively searches the hidden biases and optimal input weights of ELMs. Tenfold cross-validation (CV) is adopted meanwhile for minimizing the estimated generalization error, and then, Moore–Penrose inverse is used to obtain the output weights analytically.

Figure 1 presents the flowchart of the proposed method. The proposed JELM model consists of two parts, namely optimization of network parameters and evaluation of classification accuracy. The major aim of JELM is to achieve optimum accuracy and to restrict the input weights and bias in a predefined range to advance the convergence characteristics of the ELM. Every solution in the population of CEPO is initialized as a set of random input weights and hidden biases defined as

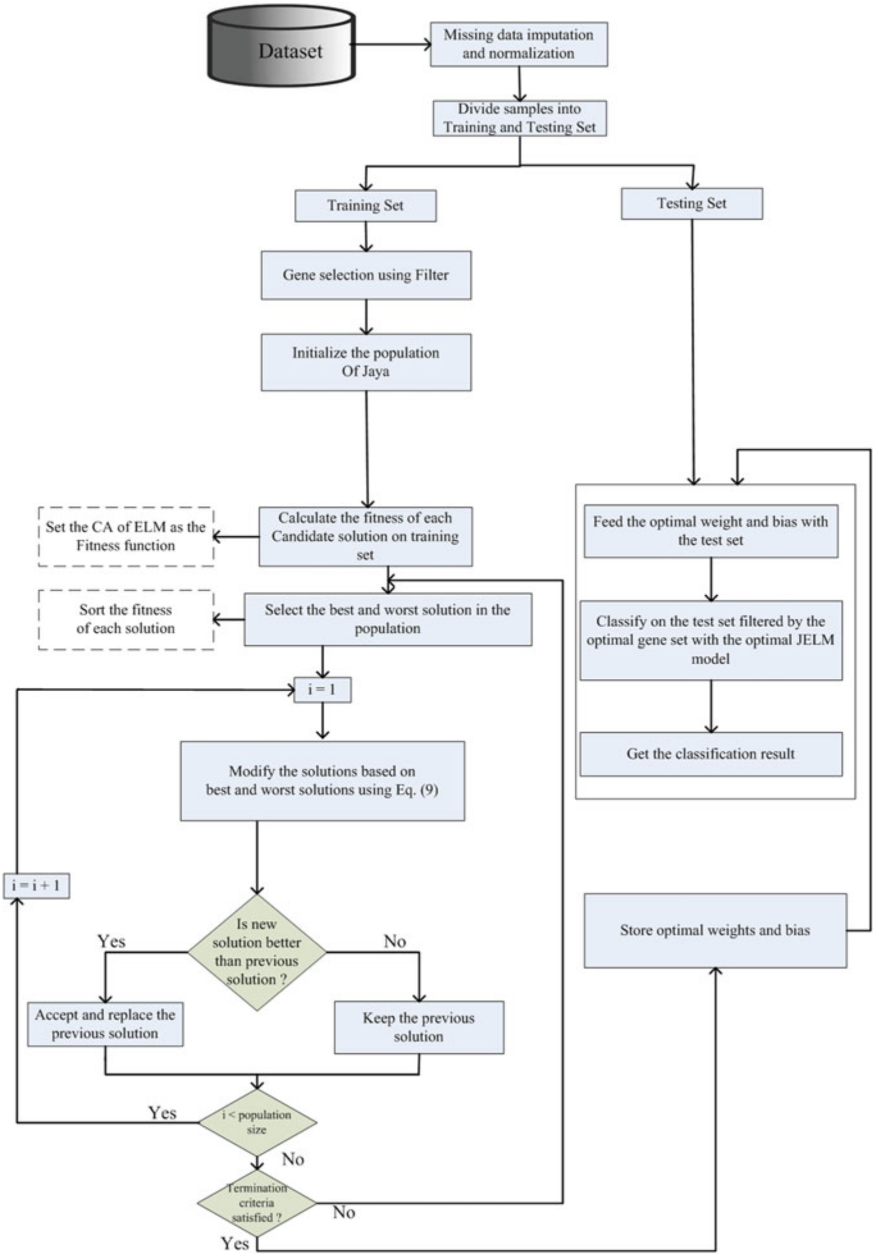


Fig. 1 Flowchart of the proposed model

$$y = [w_{11}, w_{12}, \ldots w_{1H}, w_{21}, w_{22}, \ldots w_{2H}, w_{n1} w_{n1}, w_{n2}, \ldots w_{nH}, b_1, b_2, \ldots b_H] \tag{10}$$

where

$$-1 \geq w, b \leq 1 \tag{11}$$

Then, the output weight for each solution is computed by the MP generalization inverse. The fitness value for each solution is further calculated as

$$\text{Fitness} = \text{avgACC} = \frac{\sum_{i=1}^{q} \text{testACC}_i}{q} \tag{12}$$

where avgACC indicates the mean classification accuracy of the ELM learning algorithm employing tenfold CV. q represents the size of folds.

3 Experimental Results

The whole process is carried out in four steps. Firstly, the breast cancer microarray data is extracted from a given Biomedical Dataset Repository [18]. The missing values in the dataset are filled by a missing data imputation technique, followed by normalization, in the second step. The mean value of a feature is inserted in the place of missing values, and this process is applied for every feature. Min–max normalization method is employed to normalize the values of genes to the range of [0, 1]. Again, in the third step, a gene selection method, namely Wilcoxon rank sum test, is employed to select the most appropriate genes. Finally, in the fourth step, the selected gene subset is passed through ELM classifier.

The study is conducted in a MATLAB environment on an Intel Core i5, 2.4 GHz PC with 8 GB RAM; 70, 10, and 20% of the dataset are taken for training, validation, and testing purposes, respectively, following k-fold cross-validation process. Table 1 represents a total number of genes and samples as well as some training and testing samples.

From Table 2, it is noticed that out of 24,481 breast cancer microarray genes, only 505 genes are selected from Wilcoxon rank sum test by considering the p-value as 0.001. This nonparametric statistical hypothesis test significantly decreases the gene size and selects the most relevant genes for classification.

Table 1 Dataset used

Dataset name	#Genes	#Samples	#Training samples	#Testing samples
Breast cancer	24481	97	68	19

Table 2 #Genes derived from Wilcoxon rank sum test

Dataset name	#Genes	Genes derived from Wilcoxon rank sum test
Breast cancer	24481	505

Table 3 Comparative analysis: the proposed method and benchmark classifiers

Methods	SVM	kNN	Naive Bayes	C4.5	JELM
Accuracy in %	85.19	81.48	86.89	77.78	90.91

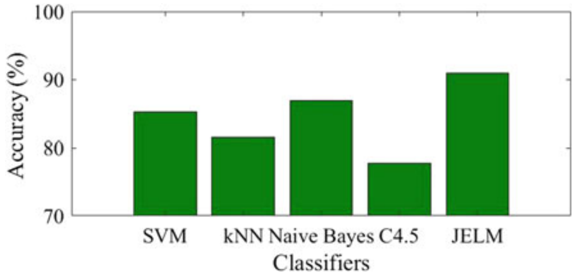


Fig. 2 Classification accuracy of proposed method with other classifiers

The performance of ELM classifier is measured according to classification accuracy. The accuracy of the ELM classifier is evaluated by its comparison with benchmark classifiers, namely SVM, *k*NN, Naive Bayes, and C4.5, and is shown in Table 3. We have noticed that ELM outperforms other classifiers in terms of classification accuracy. The obtained results are shown in terms of a bar graph in Fig. 2.

4 Conclusion

An ELM-based breast cancer microarray data analysis has been investigated in this paper. Also, a comparative analysis is carried out with other benchmark classification methods like SVM, kNN, Naive Bayes, and C4.5. From the result analysis of simulation, it is evident that a minimum subset of genes is being selected with maximum classification accuracy by the proposed method when compared to other conventional classifiers. Applications of parallel algorithms using MapReduce framework on Apache Hadoop might further improve the performance of the proposed algorithm.

References

1. Siegel, R.L., Miller, K.D., Jemal, A.: Cancer statistics, 2019. CA Cancer J. Clin. **69**(1), 7–34 (2019)

2. Flores, M., Hsiao, T.H., Chiu, Y.C., Chuang, E.Y., Huang, Y., Chen, Y.: Gene regulation, modulation, and their applications in gene expression data analysis. Adv. Bioinf. (2013)
3. Okuma, H. S., Koizumi, F., Hirakawa, A., Nakatochi, M., Komori, O., Hashimoto, J., Tamura, K., et al.: Clinical and microarray analysis of breast cancers of all subtypes from two prospective preoperative chemotherapy studies. Br. J. Cancer **115**(4), 411–419
4. Lee, G., Rodriguez, C., Madabhushi, A.: Investigating the efficacy of nonlinear dimensionality reduction schemes in classifying gene and protein expression studies. IEEE/ACM Trans. Comput. Biol. Bioinf. (TCBB) **5**(3), 368–384 (2008)
5. Xiang, Y.J., Fu, Q.Y., Ma, Z.B., Gao, D.Z., Zhang, Q., Li, Y.Y., Li, L., Liu, L., Ye, C.M., Yu, Z.G., Guo, M.M.: Screening for candidate genes related to breast cancer with cDNA microarray analysis. Chronic Dis. Transl. Med. **1**(2), 65–72 (2015)
6. Baliarsingh, S.K., Vipsita, S., Muhammad, K., Bakshi, S.: Analysis of high-dimensional biomedical data using an evolutionary multi-objective emperor penguin optimizer. Swarm Evolut. Comput. **48**, 262–273 (2019)
7. Baliarsingh, S.K., Ding, W., Vipsita, S., Bakshi, S.: A memetic algorithm using emperor penguin and social engineering optimization for medical data classification. Appl. Soft Comput. 105773 (2019)
8. Shreya, A., Vipsita, S., Baliarsingh, S.K.: Cancer classification using improved extreme learning machine. In: 2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), pp. 1–9. IEEE (2019)
9. Baliarsingh, S. K., Vipsita, S., Dash, B.: A new optimal gene selection approach for cancer classification using enhanced Jaya-based forest optimization algorithm. Neural Comput. Appl. 1–18
10. Baliarsingh, S.K., Vipsita, S., Muhammad, K., Dash, B., Bakshi, S.: Analysis of high-dimensional genomic data employing a novel bio-inspired algorithm. Appl. Soft Comput. **77**, 520–532 (2019)
11. Chuang, L.Y., Chang, H.W., Tu, C.J., Yang, C.H.: Improved binary PSO for feature selection using gene expression data. Comput. Biol. Chem. **32**(1), 29–38 (2008)
12. Sharma, A., Imoto, S., Miyano, S.: A top-r feature selection algorithm for microarray gene expression data. IEEE/ACM Trans. Computat. Biol. Bioinf. (TCBB) **9**(3), 754–764 (2012)
13. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bull. **1**(6), 80–83 (1945)
14. Liao, C., Li, S., Luo, Z.: Gene selection for cancer classification using Wilcoxon rank sum test and support vector machine. In: 2006 International Conference on Computational Intelligence and Security, vol. 1, pp. 368–373. IEEE (2006)
15. Rao, R.: Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Ind. Eng. Comput. **7**(1), 19–34 (2016)
16. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. Neurocomputing **70**(1), 489–501 (2006)
17. Zhang, R., Huang, G.B., Sundararajan, N., Saratchandran, P.: Multicategory classification using an extreme learning machine for microarray gene expression cancer diagnosis. IEEE/ACM Trans. Comput. Biol. Bioinf. (TCBB) **4**(3), 485–495 (2007)
18. Zhu, Z., Ong, Y.S., Dash, M.: Markov blanket-embedded genetic algorithm for gene selection. Pattern Recogn. **40**(11), 3236–3248 (2007)

Application of ELM-MapReduce Technique in Stock Market Forecasting

Smita Rath, Alakananda Tripathy, and Shrabanee Swagatika

Abstract Forecasting is a technique commonly used in the study of time series to forecast a variable response for a specified period of time, such as monthly earnings, stock performance, or unemployment figures. Forecasting is historical data behavior to determine the direction of future trends. Therefore, many machine learning algorithms are used in the past few years. In this study, a summary of an extreme learning machine with MapReduce technique (ELM_MapReduce) is presented. This technique is based on the concept of processing large amount of historical data and application of extreme learning machine to achieve fast learning speed. As stock market data is large set of historical data that need time to process, MapReduce method is used to handle such limitations. The technique shows the advantages and disadvantages of using MapReduce method in ELM and can be used in different areas of research.

Keywords Extreme learning machine algorithm · MapReduce techniques · Stock market

1 Introduction

ANNs have become computationally intensive for information-intensive applications with the advent of big data, reducing their large applications. The neural network's strength lies in its power to model a nonlinear system without any earlier knowledge of the process's existence [1] where the author uses a both feed forward and recurrent

S. Rath (✉) · A. Tripathy · S. Swagatika
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: smitarath@soa.ac.in

A. Tripathy
e-mail: alakanandatripathy@soa.ac.in

S. Swagatika
e-mail: shrabaneeswagatika@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_48

neural network to do forecasting. Wang et al. designed a ARIMA model to forecast the Taiwan Stock market [2]. Cao et al. have shown that ANN is an effective method for discovering the future price trends in the Shanghai Stock Exchange (SHSE) consisting of regular closing prices [3]. Moghaddam et al. [4] uses a ANN model to foresee the NASDAQ stock index using various activation functions for short-term forecasting. In traditional layers, there is a reliance between the different layers, and all the parameters are tuned to its necessity. In these above techniques, the limitations are number of hidden nodes and hidden layers to be determined, and initially, random generation of weights and bias falls prey to local minima point as the mean square error converges to zero. In today's era, main focus is given on universal estimation of compact dataset and finite learning test approximation. Therefore, extreme learning machine (ELM) comprising of one hidden layer is considered to be used in the recent field of applications. In this paper [5], the author has proved ELM over feed forward networks with and without feed backward propagation networks, ELMAN and radial basis networks. Li et al. [6] demonstrates the success of ELM over support vector machine (SVM) and radial basis function (RBF).

The paper is sorted out as follows. In Sect. 2, a basic extreme learning machine (ELM) algorithm is presented. Section 3 details of MapReduce method. Various application areas of MapReduce and extreme learning machine are highlighted in Sect. 3.1. Lastly, Sect. 4 gives the concluding section.

2 A Brief Description of Extreme Learning Machine (ELM) Technique

The ELM algorithm was originally discovered by Huang et al. [7], which is utilized as an efficient single-layer feed forward neural network for prediction of stock market indices. Extreme learning machine (ELM) appears to solve shortcomings relative to artificial neural network (ANN) [8]. ELM preserves the universal approximation power of SLFN [8–10]. ELM comprises of an input layer, single hidden layer, and output layer and uses a Moore–Penrose generalized inverse. The detailed workflow of ELM technique and Moore–Penrose generalized inverse is discussed in the next subsection.

2.1 Moore–Penrose Generalized Inverse

In linear algebra of mathematics, for a matrix B, the pseudo-inverse matrix B^+ is a generalization of the inverse matrix [11]. A popular use of the pseudo-inverse is to calculate a "best fit" solution (least squares) for a linear equation system. Penrose in the year 1956 prove that B is unique and discovered using singular value decomposition method. The following properties are to be satisfied if B^+ is the pseudo-inverse matrix of B as shown below:

$$BB^{+}B = B \tag{1}$$

$$B^{+}BB^{+} = B^{+} \tag{2}$$

$$\left(B^{+}B\right)^{\mathrm{T}} = B^{+}B \tag{3}$$

$$\left(BB^{+}\right)^{\mathrm{T}} = BB^{+} \tag{4}$$

If B is a matrix of order $m \times n$ that contains linearly separate rows, then

$$B^{+} = B^{+}(BB^{+})^{-1} \tag{5}$$

and if B is a matrix containing linearly separate columns,
then.

$$B^{+} = (B^{+}B)^{-1}B^{+} \tag{6}$$

2.2 Pseudo Code of the ELM Algorithm

The ELM consists of two phases. The first phase computes the random projection of input data, and the second phase computes the linear regression to train the weight. The algorithm of extreme learning machine is depicted as below:

In the given input, series training set $S = \{(x_i, t_i)\}$ for $i = 1 \ldots n$ and the activation function in the network is sigmoid as stated in relation (7)

$$F(w_i, b_i, x_i) = \frac{1}{1 + \mathrm{e}^{-(w.x+b)}} \tag{7}$$

where w_i are the set of random weights and b is the bias in the input-hidden layer.

Step 1: The weights and bias in the first phase are considered random.
Step 2: Compute the hidden matrix H_{11} according to relation (8)

$$H_{11} = \begin{bmatrix} F(w_1,\ b_1,\ x_1) \ldots F(w_m,\ b_m,\ x_1) \\ \ldots \qquad \ldots \qquad \ldots \\ F(w_1,\ b_1,\ x_n) \ldots F(w_m,\ b_m,\ x_n) \end{bmatrix} \tag{8}$$

Step 3: Find the output weight from Eq. (9) using Moore–Penrose generalized inverse.

$$\beta = H_{11}^{+} T \tag{9}$$

Step 4: Exit

3 MapReduce in Reducing the Dimensionality of Stock Data

Even though ELM is with the benefit of rapid training speed, in the face of large sample data, the efficiency and scalability will still be the big challenges as the large sample amount along with the large number of hidden nodes usually leads to a complex H matrix calculation. Therefore, the MapReduce technique, a current cloud computing method, is used to overcome the limitation in use of big data as input data in the ELM network. The map function takes data from the input and produces pairs of intermediate key and value. Then, the reduction function considers an intermediate key with a set of values to construct a smaller set of values. The reducer usually only generates zero or one output value. MapReduce technique proves to be an efficient way to solve the dimensionality and scalability problem in big data [12]. Chen et al. in [13] proposes a MR-ELM model to handle the high dimension dataset and enable large-scale ELM workout. Liu et al. in their paper [14] introduce a parallelized BPNN focused on the MapReduce computing model which offers additional features such as replication of data and fault tolerance. Venkatraman designed a new MapReduce CBIR neural network architecture from large cloud data collection, follows natural language question to identify color pictures, and uses map reduction method that forms cloud clusters to achieve accurate outputs in real time [15]. Namitha in [16] proposes a rainfall prediction model that comprises of a generalized feed forward neural network and builds a MapReduce model to handle the rainfall data. The whole process of MapReduce consists of three phases mapping, reducing, and combining which is depicted in Fig. 1.

- **Mapping Phase**—The input data is split into smaller datasets using the map function. The map function basically consists of a value and key which process each data sample from splitting process.
- **Reducing Phase—**This phase does the shuffling of the data that are output from the mapping phase and then performs merging of data that have same key values. Then, the data with the same key value pairs are grouped together, and sorting is done to arrange the list of data based on key values.
- **Combining Phase—**The output list of data obtained in the previous phase is gathered and combined to form the new dataset to be taken as input to the ELM network.

Table 1 illustrates the advantages and disadvantages of the MapReduce technique.

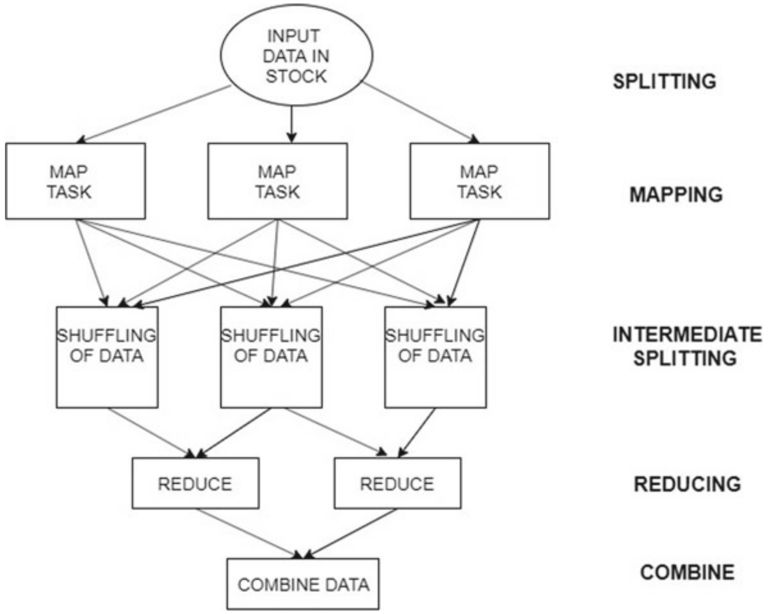


Fig. 1 Phases of MapReduce framework

Table 1 Advantages and limitations of MapReduce technique

Advantages
• It is highly scalable
• Large data are stored over parallel with more processing speed
• Requires very cheap data storage and processing. It can be used later on as well
• In reality, cost savings are huge, and costs for each terabyte of data can be reduced from thousands and percentages to hundreds
• Flexibility in accessing different datasets in areas such as market analysis, fraud detection, and data warehousing.
• Fast processing speed
• Divides the whole set of data into smaller subsets to run in less time
• Easy to code using simple programming language
Disadvantages
• Have to wait for all the parallel working task to complete to go to the next step.
• Can be used only in batch processing

Table 2 Performance analysis of the working model

Result analysis	NASDAQ
MSE (maximum) $(\times 10^{-4})$	2.2988
MSE(mean) $(\times 10^{-4})$	2.4276
MSE (minimum) $(\times 10^{-4})$	2.3223
MAPE (maximum) $(\times 10^{-4})$	2.8340
MAPE(minimum) $(\times 10^{-4})$	1.5541
MAPE (mean) $(\times 10^{-4})$	1.6980

3.1 Result Analysis

Simulation is carried on NASDAQ composite dataset for ten years historical data. Initially, a MapReduce method is used to pre-process the data and then fed into the extreme learning machine. The inputs are open, low, high, prev close of the dataset. A random number of weights and bias is generated, and the network is run 3000 times. Performance of the network is shown in Table 2 in terms of mean square error (MSE) and mean percentage absolute error (MAPE). Figure 2 shows the predicted stock price verses actual price and determines the closeness of prediction.

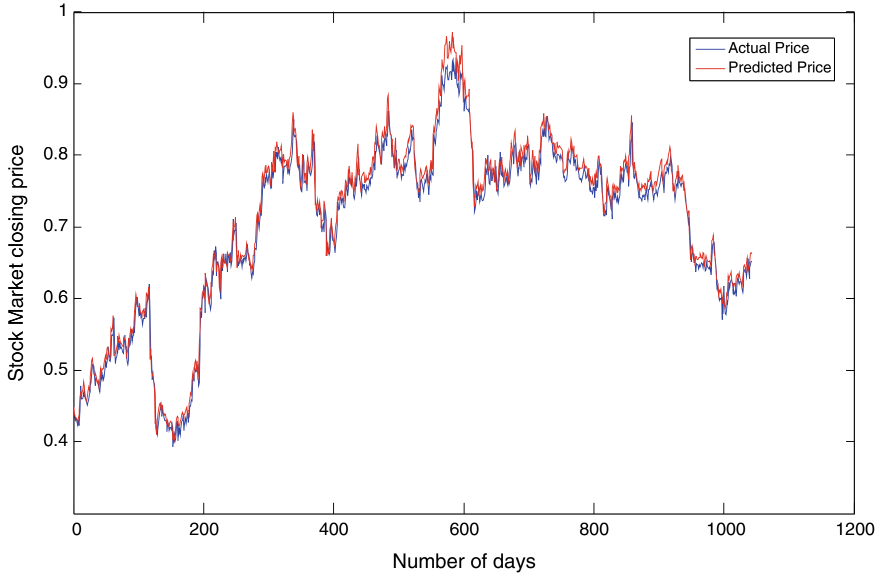


Fig. 2 Predicted price verses actual price of NASDAQ stock index

4 Conclusions

A large number of people are finally paying attention to the future market with the increase in living styles and the shift in financial thought. Inorder to raise profits and reducing risk with the evolving stock indices, investors need to know the price in advance and know the price in advance in order to adjust the investment timing rationally. A machine learning technique works more effectively compared to the other prediction algorithm when compared with other neural network schemes. For stock market forecasting, which relies on the selected input data, several forecasting algorithms are available nowadays. Hadoop's MapReduce technology enables the storage of such huge volumes of information in an entirely safe and price-effective manner when it comes to processing large sets of information. This paper presents the effectiveness of MapReduce method in handling large datasets and its use and limitations. Extreme learning machine has proved its efficiency in many fields like regression and classification problems.

References

1. Khoa, N.L.D., Sakakibara, K., Nishikawa, I.: Stock price forecasting using back propagation neural networks with time and profit based adjusted weight factors. In: 2006 SICE-ICASE International Joint Conference, pp. 5484–5488. IEEE (2006)
2. Wang, J.H., Leu, J.Y.: Stock market trend prediction using ARIMA-based neural networks. In: Proceedings of International Conference on Neural Networks (ICNN'96), vol. 4, pp. 2160–2165. IEEE (1996)
3. Cao, Q., Leggio, K. B., Schniederjans, M.J.: A comparison between Fama and French's model and artificial neural networks in predicting the Chinese stock market. Comput. Oper. Res. **32**(10), 2499–2512 (2005)
4. Moghaddam, A.H., Moghaddam, M. H., Esfandyari, M.: Stock market index prediction using artificial neural network. J. Econ. Finan. Admin. Sci. **21**, 89–93 (2016)
5. Sivalingam, K.C., Mahendran, S., Sivanandam Natarajan, S.: Forecasting gold prices based on extreme learning machine. Int. J. Computers. Commun. Control **11**: 372–380 (2016)
6. Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., Deng, X., et al.: Empirical analysis: stock market prediction via extreme learning machine. Neural Comput. Appl. **27**(1), 67–78 (2016)
7. Huang, G., Huang, G.B., Song, S., You, K.: Trends in extreme learning machines: a review. Neural Netw. **61**, 32–48 (2015)
8. Huang, G.-B., Chen, L.: Convex incremental extreme learning machine. Neurocomputing **70**(16–18): 3056–3062 (2007)
9. Huang, G-B.: An insight into extreme learning machines: random neurons, random features and kernel. Cogn. Comput. **6**:376–390 (2014)
10. Huang, G.-B.: What are extreme learning machines? Filling the gap between Frank Rosenblatt's dream and John von Neumann's puzzle. Cogn. Comput. **7**, 263–278 (2015)
11. Rakha, M.A.: On the Moore-Penrose generalized inverse matrix. Appl. Math. Comput. **158**, 185–200 (2004)
12. Jeffrey, D., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Commun. ACM **51**, 107–113 (2008)
13. Chen, J., Chen, H., Wan, X., Zheng, G.: MR-ELM: a MapReduce-based framework for large-scale ELM training in big data era. Neural Comput. Appl. **27**(1), 101–110 (2016)

14. Liu, Y., Xu, L., Li, M.: The parallelization of back propagation neural network in mapreduce and spark. Int. J. Parallel Prog. **45**(4), 760–779 (2017)
15. Venkatraman, S., Kulkarni, S.: MapReduce neural network framework for efficient content based image retrieval from large datasets in the cloud. In. 2012 12th International Conference on Hybrid Intelligent Systems (HIS), pp. 63–68, IEEE (2012)
16. Namitha, K., Jayapriya, A., Kumar, G. S.: Rainfall prediction using artificial neural network on map-reduce framework. In. Proceedings of the Third International Symposium on Women in Computing and Informatics, pp. 492–495, ACM (2015)

FTTCR: A New Interconnection Architecture for Parallel Systems

Laxminath Tripathy and Asif Uddin Khan

Abstract Mesh and Torus are the most popular interconnection topologies. Based upon Fully Twisted Torus, a new interconnection topology is proposed and compared with some existing topologies. It is found that the proposed topology has better topological parameters than TCR, FTT, and other networks under consideration. The Fully Twisted Torus Connected Ring is an enhanced TCR interconnected network. The topological properties of the proposed network are analyzed.

Keywords TCR · FTT · Interconnection networks

1 Introduction

A torus interconnected is a network topology for connecting processing nodes in a parallel computer system. It can be visualized as a mesh interconnected with nodes arranged in a rectilinear array of $N = 2$, 3 or more dimensions with processors connected to their neighbours, and the corresponding processor on opposite edges of the array connected [1]. Torus and Mesh topologies are also found in most of the commercial architecture, like the Alpha 21364 (two-dimensional Torus), that are mainly targeted at the application domains such as database servers, web servers, and telecommunication [2]. The torus interconnection network is frequently used on top-performing Supercomputers. The organization of this paper is as follows. In Sect. 1: Introduction, Sect. 2: Related work Sect. 3: Proposed topology. Section 4: Topological properties. Section 5: Comparison. Section 6: Conclusion has been mentioned.

L. Tripathy (✉)
Siksha 'O' Anusandhan University (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: laxminathtripathy@soa.ac.in

A. U. Khan
Department of CSE, Silicon Institute of Technology, Bhubaneswar, India
e-mail: asif.khan@silicon.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_49

2 Related Work

Torus network has a ring connection along each row and each column. A torus interconnected is a network topology for connecting processing nodes in a parallel computer system. Mesh network is a simple network for the general-purpose applications. But as the size of the Mesh increases, the network performance degrades dramatically due to the large network diameter and little bisection width [3]. The DMesh and DTorus networks are proposed in [4] to enhance the performance and scalability of the both Mesh and Torus network. To satisfy the special need of the Network-On-Chip (NoC), a Mesh-like topology named XMesh, and its routing algorithm called XM are presented in [5]. The SD-Torus network [6] is a regular and symmetrical interconnection network. CCtorus [7] is an improved variation of a torus and another improvement of Torus called TM topology [8], has less number of links than the torus. This TM topology is better than torus in terms of node degree, number of links. Its diameter is less than mesh topology and equal with torus topology. Authors [9] also proved that TM topology performed better than other topologies under different traffic patterns. Similar to a torus, authors in [10] discussed another topology called Hierarchical Interconnection Network-on-Chip (HCCR). It has better topological parameters than Mesh. It has less degree and less number of links than Mesh.

There are other topologies like DiaTorus [11], Dmesh [12], Diametric mesh [13], Tmesh [14], xmesh, xtorus and xxtorus [15], CBP-Mesh and CBP Torus [16, 17] have no uniform degree and the number of links is also more. The new torus network, which has the hypercube Q3 as the basic module has been proposed [18]. The proposed Hypertorus has degree 4, and its network has the node and its edge symmetric and is scalable. The extra links and ports require more arbitration levels and complex routing algorithms.

Our proposed topology is compared with the above topologies and shown that it is better, simple in design and each router has a uniform degree. A system with a small node degree that implies less hardware cost per node and a constant node degree implies that the system can be scalable without having to modify the individual nodes [19].

3 Proposed Topology

The proposed topology is similar to TCR [20] but the wraparound connections are in a twisted form in both X and Y direction. Each node position is represented by three coordinates. The first and second coordinates represent the position of nodes called supernode along x and y direction respectively. The third coordinate represents the position of a node in the ring. The connection pattern of a supernode $A(x_1, y_1)$ in FTTCR for dimension $n \times n$ is as follows:

1. If x_1 and $y_1 != n - 1$, the four connected super node positions are

$$B(x_1+1, y_1), C(x_1, y_1+1), D(x_1-1, y_1), E(x_1, y_1-1).$$

2. If position of A is $(0, y_1)$

$$B(n-1, t_1), C(1, y_1), D(0, y_1+1), E(0, y_1-1)$$

3. If position of A is at $(x_1, 0)$, A is connected to following supernodes

$$B(t_2, n-1), C(x_1+1, 0), D(x_1-1, 0), E(x_1, y_1+1)$$

4. If position A is (0,0) then the connected supernodes are in the following positions

$$B(t, n-1), C(n-1, t), D(1, 0), E(0, 1).$$

The value of t_1 in cases of $2 = \text{Mid} + y_1 \bmod n$, Similarly, the value of t_2 in case $3 = \text{Mid} + x_1$ and in case 4 value of $t = \text{Mid}$ where Mid is the middle position and mod is mathematical module operation

The FTT [21] is shown in Fig. 1 for and Fig. 2 depicts the TCR network. The wraparound connections are only in the boundary position. For Example, FTT for $n = 5$, the wraparound connection in x dimension is as follows For position 0 in the X direction is wraparound with $(0 + 3) \bmod 6 = 3$ 0 is connected to $3 = 0$–3. Similarly 1–4, 2–4, 3–5, 4–0, 5–2 along both X and Y direction.

FTTCR can be obtained from FTT by changing the connection patterns of TCR. In each position, the ring is called a supernode. So in connection pattern, only supernodes are considered as within the ring four nodes are directly connected in a cyclic manner (Fig. 3).

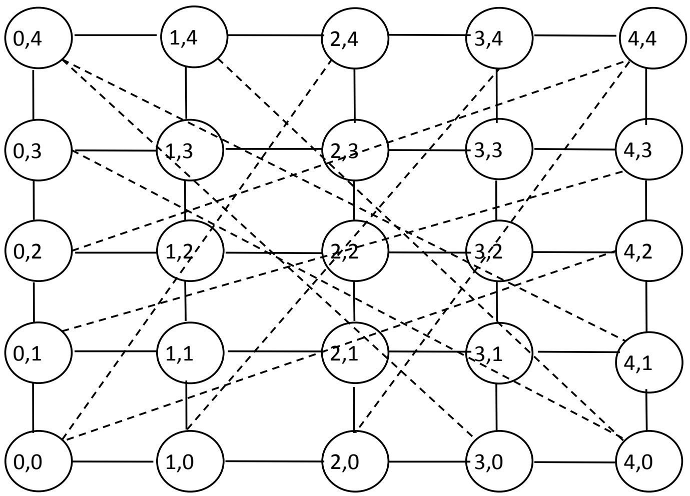


Fig. 1 Fully twisted torus (FTT) for $n = 5$

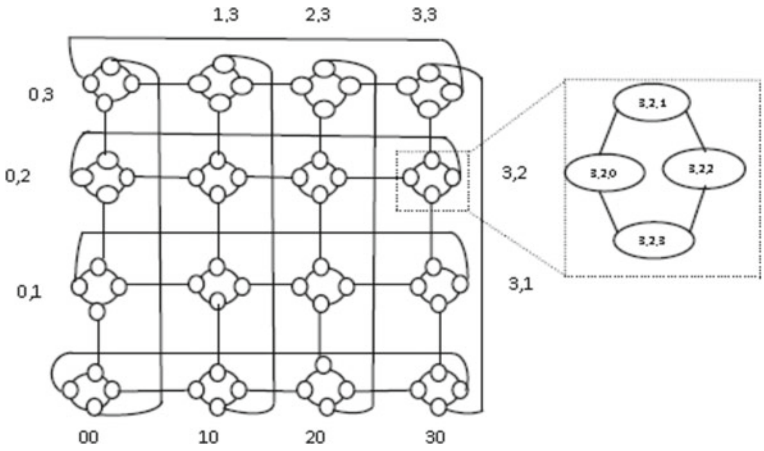


Fig. 2 Torus connected ring (TCR)

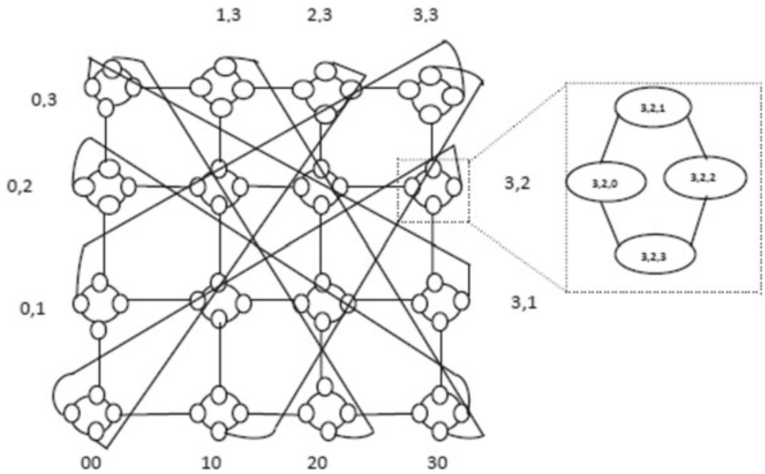


Fig. 3 Fully twisted torus connected ring (FTTCR) for $n = 5$

4 Topological Properties of FTTCR

The degree of FTTCR is three irrespective of dimension. FTTCR topology has the following important topological properties (Assume the number of nodes $= 4n^2$) as follows

Property 1 FTTCR topology is a regular and the node degree is 3. Due to the uniformity in degree, the design of router complexity is less. The hardware cost per node of topology is directly related to node degree and a constant node degree indicates that the system is scalable without modifying the individual nodes [6].

Property 2 The total number of links present in FTTCR topology is $6n^2$. The number of links that connect each ring is $2n^2$, each ring has 4 links. So, the total of n^2 ring and the total number of links is $4n^2 + 2n^2 = 6n^2$.

Property 3 The scalability of FTTCR topology is high. The scalability is defined as the property of expanding the network size, getting a constant performance. The scalability of FTTCR can be achieved by adding the same number of nodes in both x and y direction. For example, l number of rings can be added in both x and y direction of an $m \times m$ FTTCR topology, and the new structure will be extended $(m + l) \times (m + l)$ FTTCR topology. After adding new nodes to the connection patterns and links remain the same.

Property 4 The network diameter is defined as the maximum of all the shortest paths between any two nodes. For example, a ring of 4 nodes has diameter 2 and the diameter of the Fully Twisted Torus (FTT) is $n - 2$. The diameter of FTTCR can be computed from the addition of the diameter of the FTT and the product of the diameters of both FTT and ring. Hence, the diameter of FTTCR is $(n - 2) + 2 \times (n - 2) = 3n - 6$. The FTT is a network with a constant node degree and is highly scalable architecture, and it has a smaller network diameter in comparison to mesh.

Property 5 The bisection width of a network is the minimum number of links whose removal disconnects the graph and cuts it in two halves. The bisection width of FTTCR is $3n$.

Property 6 The average distance of a FTTCR network topology structure is the sum of average distances of both FTT and ring. The average distance of each ring is $4/4 = 1$ and the $n \times n$ FTT structure is $n/2$. So the average distance of FTTCR is $(n/2 + n/2 \times 1) = n$. As the average distance of MCR is $4n/3$ so, latency is more than FTTCR.

Property 7 The cost of a network is the scalar product of the degree and diameter. As the degree and diameter of FTTCR are 3 and $3n – 6$, respectively, so the cost is $9n - 18$.

Property 8 The packing density of a network is defined as the total number of nodes per total network cost. It indicates the size of the chip area of VLSI layout. The larger package density indicates the smaller chip area of VLSI design layout which is also a desirable property. As the size of FTTCR is N and cost $= 9n - 18$, so packing density is $N/(9n - 18)$, where $N = 4n^2$.

5 Comparison with Other Topology

The number of links and degrees of FTTCR is the same as of Torus Connected Ring but its diameter is $n - 2$ which is less than torus and all other topologies under consideration. Due to the small network diameter, the latency will be less in comparison to Mesh, MCR, and TCR. As the degree is the same and the diameter of topology is less than TCR, Torus, and MCR. The topological analysis is shown in Figs. 4, 5 and 6. This analysis is obtained from the topological properties of different networks mentioned in Table 1.

The cost of the topologies can be obtained from the scalar product of the degree of router and diameter. The cost graph is shown in Fig. 5. It depicts that the cost of FTT is the lowest among Mesh, Torus, and Mesh.

The packing density should be more for designing a network-on-chip. The more packing density is more suitable for the selection of network-on-chip design. The

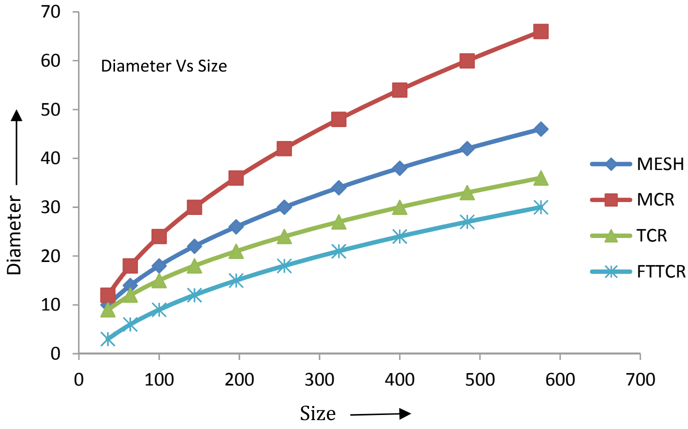


Fig. 4 Diameter analysis

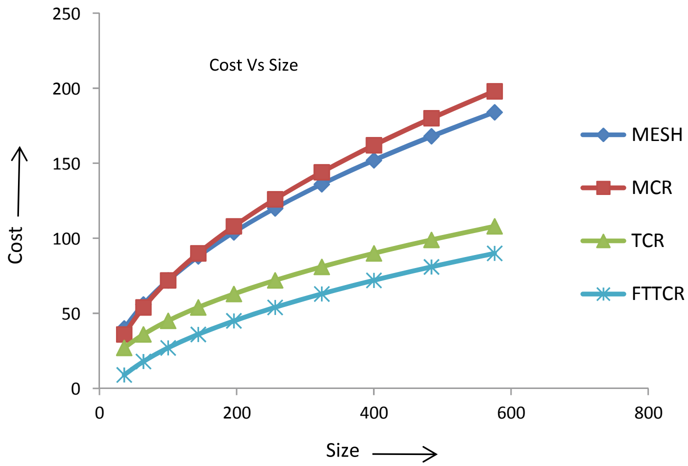


Fig. 5 Cost analyses

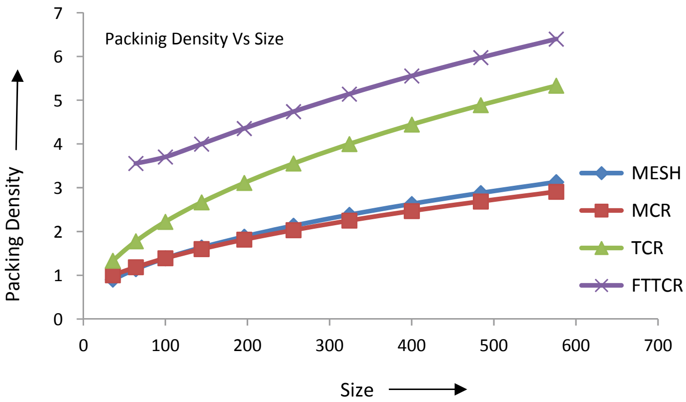


Fig. 6 Packing Density Analyses

Table 1 Topological parameters comparison

Topology	Node degree	Average distance	Network diameter	Bisection width	Cost	Size	Packing density
Mesh	4	$4n/3$	$2(2n-1)$	$2n$	$8(2n-1)$	$2n \times 2n$	$n^2/2(2n-1)$
MCR	3	$4n/3$	$6(n-1)$	n	$18(n-1)$	$n \times n$	$2n^2/9(n-1)$
TCR	3	n	$3n$	$2n$	$9n$	$n \times n$	$4n^2/9n$
FTTCR	3	n	$3n-6$	$3n$	$9n-18$	$2n \times 2n$	$4n^2/(9n-18)$

analysis of the packing density depicted in Fig. 6. From this figure, it is clearly shown that the packing density of FTTCR is the highest among other topologies.

6 Conclusion

This proposed structure called Fully Twisted Torus Connected Ring is regular, flat, and highly scalable. After the topological analysis of MCR, TCR, and Mesh with FTTCR, it is observed that FTTCR has good topological parameters. It has multiple paths between any source and destination pair for which it has better fault tolerance capability than TCR. The cost of topology is less than mesh, TCR, and Torus. FTTCR topology has a better trade-off between the cost of the network and its performance. The design of this network is simple. The main beautiful feature of this topology is that it has fewer diameters than all topologies under consideration and uniform degree which simplifies design complexity.

References

1. Koniges, A.E.: Industrial Strength Parallel Computing. ISBN 1990
2. Mukherjee, S.S., Bannon, P., Lang, S., Spink, A., Webb, D.: The alpha 21364 network architecture. IEEE Micro **22**(1), 26–35 (2002)
3. Arora, L.K,: 'C^2Mesh' interconnection network topology based on 2D mesh. In: 3rd IEEE International Advance Computing Conference (IACC) (2013). 978-1-4673-4529-3/12
4. Ouyang, Y.M., Zhu, B., Liang, H.G., et al.: Networks on chip based on diagonal interlinked mesh topology structure. Comput. Eng. **35**(22), 100–102 (1999)
5. Zhu, X.J., Hu, W.W., Ma, K., et al.: XMesh: mesh like topology for network on chip. J. Softw. **18**(9), 2194–2204 (2007)
6. Wang, Y., Du, H., Shena, X.: Topological properties and routing algorithm for semidiagonal torus networks. J. China Univ. Posts Telecommun. **18**(5), 64–70
7. Bhardwaj, M.: C^2Torus New interconnection network topology based on 2D Torus. Am. J. Netw. Commun. Special Issue: Ad Hoc Netw. **4**(3–1), 1–4 (2015). https://doi.org/10.11648/j.ajnc.s.2015040301.11
8. Wang, X., Xiang, D., Yu, Z.: TM: a new and simple topology for interconnection networks. J. Supercomput. **66**(1), 514–538 (2013)
9. Moudi, M., Othman, M.: Mathematical modelling for TM topology under uniform and hotspot traffic patterns. AUTOMATIKA **58**(1), 88–96 (2017)
10. Inam, O., Al Khanjari, S., Vanderbauwhede, W.: (2015) Shortest path routing algorithm for hierarchical interconnection network-on-chip. Procedia Computer Sci. **56**, 409–414 (2015)
11. Thakyal, D., Chatterjee, P.: Dia-Torus: a novel topology for network on chip design. Int. J. Comput. Netw. Commun. (IJCNC) **8**(3), 137–148 (2016)
12. Wang, C., Hu, W.-H., Lee, S.E., Bagherzadeh, N.: Area and power-efficient innovative congestion-aware network-on- chip architecture. J. Syst. Archit. 24–38 (2011)
13. Reshadi, M., Khademzadeh, A., Reza, A., Bahmani, M.: A novel mesh architecture for on-chip networks. Des. Reuse Ind. Art. (2013)
14. Yang, Q., Wu, Z.: An improved mesh topology and its routing algorithm for NoC. In: Proceedings of International Conference on Computational Intelligence and Software Engineering (CiSE), pp. 1–4 (2010)
15. Yu-hang, L., Ming-fa, Z., Jue, W., Li-min, X., Tao, G.: Xtorus: an extended torus topology for on-chip massive data communication. In: Proceedings of IEEE 26th International Symposium Workshops & Ph.D. Forum on Parallel and Distributed Processing, pp. 2061–2068 (2012)
16. Gulzari, U., Anjum, S., Agha, S.: Cross by pass-mesh architecture for on-chip communication. In: 9th International Symposium (2015)
17. Gulzari, U.A., Sajid, M., Anjum, S., Agha, S.: Torres FS a new cross-by-pass-torus architecture based on CBP-mesh and torus interconnection for on-chip communication. PLoS ONE **11**(12), e0167590 (2016). https://doi.org/10.1371/journal.pone.0167590
18. Ki, W., Lee, H.-O., Oh, J.C.: The new torus network design based on 3-dimensional hypercube. In: Proceedings of 11th International Conference on Advanced Communication Technology, ICACT 2009, vol. 01 (2009)
19. Louri, A., Sung, H.: An optical multi-mesh hypercube: a scalable optical interconnection network for massively parallel computing. IEEE J. Light wave Technol. **12**, 704–716 (1994)
20. Tripathy, L., Tripathy, C.R.: A new interconnection topology for network on chip. Int. J. Comput. Netw. Commun. (IJCNC) **10**(4), 37–50 (2018)
21. Tripathy, L., Tripathy, C.R.: FTT: a new interconnection topology. In: 2nd International Conference on Innovative Trends in Engineering, Applied Science and Management. (ICITEASM-2018), pp. 179–182 (2018)

Risk Prediction of Kidney Disease Using Machine Learning Strategies

Lambodar Jena, Bichitrananda Patra, Soumen Nayak, Sushruta Mishra, and Sushreeta Tripathy

Abstract Classification is the most commonly applied machine learning technique that classifies large population of records based on the training set and the feature values. The important task of a classifier is to predict categorical class labels and construct a model for the target class. The classification techniques are widely used in the emerging research fields of bioinformatics. Prediction of disease that is chronic in nature is a big challenge for medical experts. Thus, in the field of bioinformatics, it is vital to predicting the disease accurately that will help the physicians to begin the treatment process. This work develops a chronic disease prediction model by implementing various machine learning classification techniques. The model analyzes the data from the data set and results in the prediction accuracy in each case of the classifier.

Keywords Machine learning · Classification · Prediction · Feature selection · Genetic search

1 Introduction

Day by day large amount of data is collected due to the rapid growth of technology and science. Particularly in medical science huge amount of medical databases are created to assist the healthcare system [1]. Now a major field of research is the management of large scale of data and efficient extraction of knowledge from large databases [2]. In the recent era, machine learning has become an emerging field to

L. Jena (✉) · B. Patra · S. Nayak
Department of Computer Science & Engineering, Siksha ′O′ Anusandhan (Deemed to be) University, Bhubaneswar, Odisha, India
e-mail: lambodarjena@soa.ac.in

S. Mishra
School of Computer Engineering, KIIT University, Bhubaneswar, Odisha, India

S. Tripathy
Department of Computer Science & Information Technology, Siksha ′O′ Anusandhan (Deemed to be) University, Bhubaneswar, Odisha, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_50

be used on bioinformatics and complex studies of medical science. Large volumes of data are yet to be explored and mined to gather knowledge. The main focus of this work is to process the chronic kidney disease dataset for the prediction of chronic kidney disease of a large population. Based on the symptoms identified in patients it predicts the sick ones [5]. In this work, genetic search algorithm is used for feature selection and four machine learning classifiers are used for classification. The classifiers are namely Naïve Bayes, Multilayer Perceptron, J48, and Decision Table. For better prediction and diagnosis of chronic kidney disease first, the classifiers are used and compared with each other. Then subsequently the features are reduced from the dataset by using a genetic search algorithm and again classification results of each classifier are compared with each other.

2 Related Work

Many authors have implemented different machine learning techniques for classification and future prediction. Dhamodharan [1] predicted risk in liver disease by using a number of machine learning classifiers and he found that Naïve Bayes becomes suitable algorithm with high prediction accuracy. Solanki [2] has used J48 and Random tree algorithms for the classification of sickle cell disease and found that Random tree performs better. Breast cancer diagnosis and prognosis have been done by Joshi et al. [3] using some rules of classification. They found that LMT (logistic model tree) algorithm identifies more correctly (76%) the healthy and sick patients. However, the disease leukemia risk has been predicted by David et al. [4]. They have implemented a number of classifiers like KNN, J48, Random tree and Bayesian n/w and found that Bayesian performs well. Heart disease risk prediction has been done by Vijayarani and Sudha [5] by using different types of classifiers. Kumar [6] has used alternating decision trees for dengue fever diagnosis. In the healthcare sector, Durairaj and Ranjani [7] have used and compared the efficiency of various types of data mining algorithms. They have worked on the prediction and diagnosis of life-killing diseases. Sugandhi et al. [8] used weka to analyze a cataract patient's database. Yasodha and Kannan [9] analyzed the diabetic patient database using different methods namely Network, Bayes Tree, J48 and Random Tree. Different classification techniques such as Radial Basis function, Bayes Network, Decision Tree, and pruning algorithms are compared by Bin and Yau [10] for breast cancer. Jena and Kamila [11] worked on chronic kidney disease dataset by comparing different classification techniques such as SVM, Naïve, Conjunctive rule, MLP, J48, and Decision Table. Also, Jena and Ramakrushna [12] further worked on the prediction of kidney disease using different algorithms and feature selection techniques. Jena and Kamila [13] have also worked on the prediction of human depression using apriori algorithm.

3 Background

There are two forms of data analysis namely classification and prediction. They are used for extracting models describing important classes or to predict future data trends. Classification models predict categorical class labels; and prediction models predict continuous-valued functions.

There are two steps for Data Classification process:

- Classification Model building (Classifier)
- Classification.

Model Building or classifier Building

- This is the learning phase.
- The classifier is build-up by classification algorithms.
- The training set and their associated class labels are used to build the classifiers shown in Fig. 1.

Classification Using Classifier

Here, classification is done by the classifiers. The test data is used to estimate the accuracy level of classification rules. The classification rules can be applied to the new data tuples if the accuracy level is considered acceptable shown in Fig. 2.

Attribute Selection

It is a mechanism by virtue of which the most relevant features or attributes are selected for a predictive model. This method identifies and removes unneeded, redundant, and irrelevant attributes or features that do not contribute to or decrease the accuracy of the predictive model. In this regard, genetic search algorithm is a popular one.

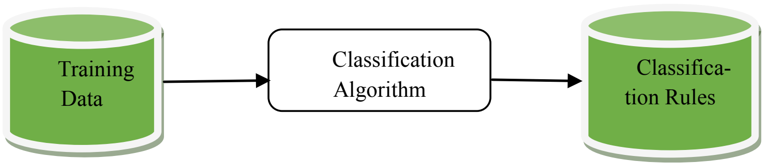


Fig. 1 Building of classifier

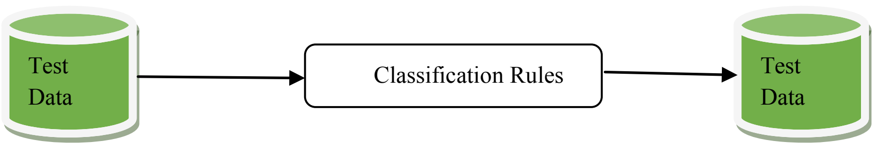


Fig. 2 Classification

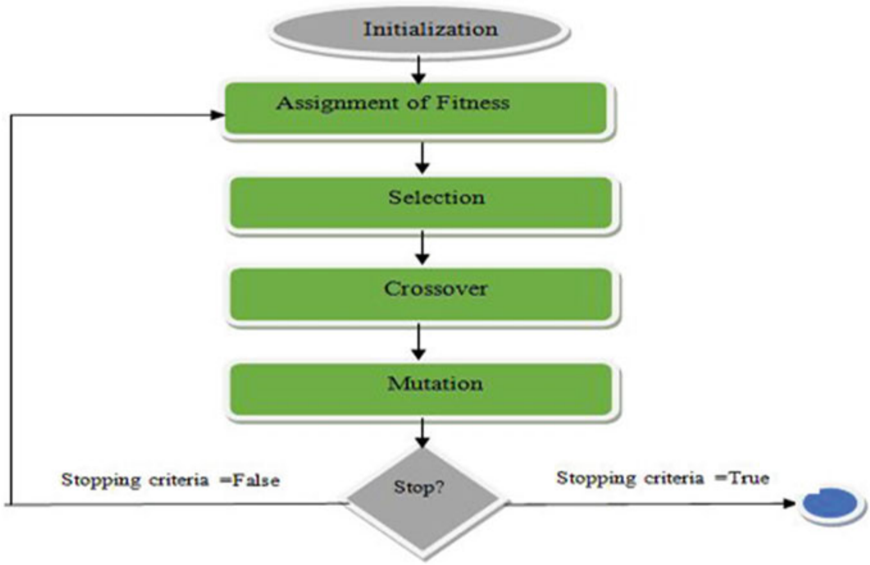


Fig. 3 Feature selection process using genetic algorithm

In our work, we have used genetic algorithm method for feature selection. It is a stochastic method for optimizing the functions that are inspired by natural genetics and biological evolution. The working process of genetic algorithm is depicted in Fig. 3.

4 Result and Discussion

The experiment is carried out on the chronic kidney disease dataset which has 400 instances and 24 attributes. The resulted values of different performance indicators are depicted in Tables 1 and 2. Table 1 contains values resulted by the classifiers considering all the attributes of the dataset and Table 2 contains the classification results of all parameters after feature reduction using genetic search algorithm. Figures 4, 5, 6, 7, 8, 9, 10, 11, and 12 show the graphical analysis of each parameter's against each classifier used in both the experiments.

From Table 1, it is clearly visible that Multilayer Perceptron classifier obtains better results than the other algorithms in terms of the accuracy (= 99.75%), Kappa statistics (= 0.99471), MAE (= 0.008511), RMSE (= 0.06221), and ROC (= 1). Since it gives more accuracy for prediction of the chronic kidney disease it takes more time to build the model. The model building time of MLP is 8.631 s which is high in comparison to other algorithms. However, its model testing time (= 02 s) on the same training data remains the same with Naïve Bayes algorithm which gives 95% accuracy, and Decision Table algorithm which gives 99% accuracy. In the case

Table 1 Classification results before feature reduction

Algorithm	% Of accuracy	Kappa statistics	MAE	RMSE	ROC	Model building time	Model testing time on training data
Naïve Bayes	95	0.89611	0.047912	0.20461	1.0	0.021	0.02
Multilayer perceptron	99.75	0.99471	0.008511	0.06221	1.0	8.631	0.02
J48	99	0.97861	0.022521	0.08071	0.999	0.081	0
Decision table	99	0.97862	0.181521	0.25071	0.992	0.221	0.02

Table 2 Classification result after feature reduction using genetic search

Feature reduction	Classification algorithms	%Of accuracy	Kappa statistics	MAE	RMSE	ROC	Model building time	Model testing time on training data
Genetic search	Naïve Bayes	97	0.93701	0.0334	0.1608	1.00	0.01	0.01
	Multilayer perceptron	99.75	0.99471	0.0082	0.061	1.00	3.81	0.02
	J48	99	0.97861	0.0225	0.0805	0.999	0.01	0.01
	Decision table	98.75	0.97321	0.163	0.2331	0.99	0.08	0.01

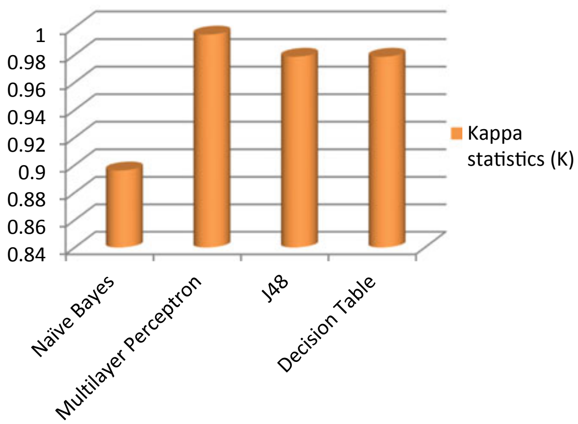


Fig. 4 Analysis of Kappa statistics (K)

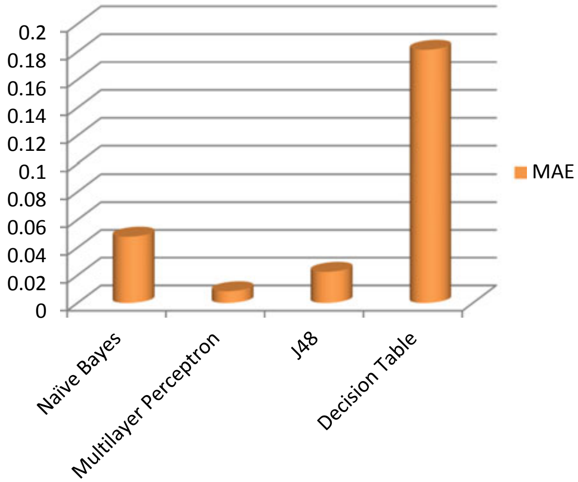
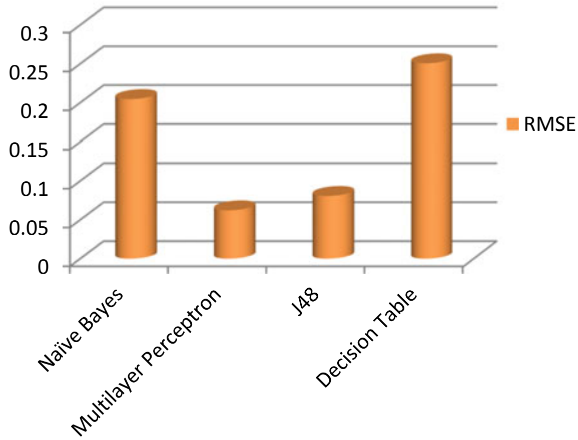


Fig. 5 Mean absolute error (MAE) analysis

Fig. 6 Analysis of RMSE



of MLP the error rate value, i.e., the MAE value and RMSE value are 0.008511 and 0.06221, respectively, which is minimum in comparison to the other two algorithms. Thus, its prediction result is very near to the actual value. The ROC value (= 1) of MLP is the same as the Naïve Bayes but very close to J48 (= 0.999) and Decision Table (0.992) algorithm. The performance of all these classification algorithms with respect to the above criteria is depicted in Figs. 4, 5, 6, 7, 8, and 9.

The above discussion is based on without using feature selection. But the following discussion is dealt with using genetic search feature selection. The genetic search with correlation-based feature selection (CFS) is a subset evaluator (an attribute subset evaluator) and the same dataset taken for without using feature selection is tested again. The genetic search evaluates all training data initially. The result of feature

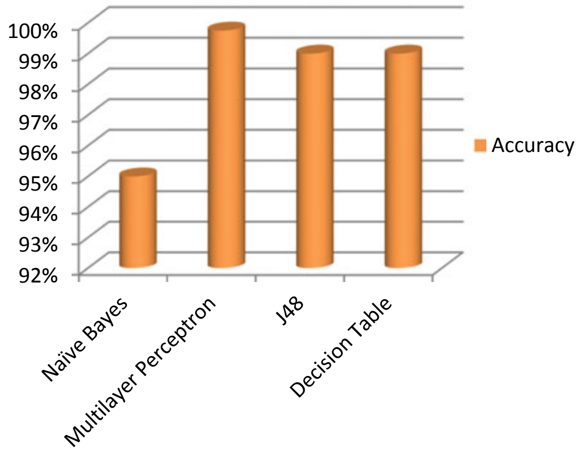


Fig. 7 Classification accuracy analysis

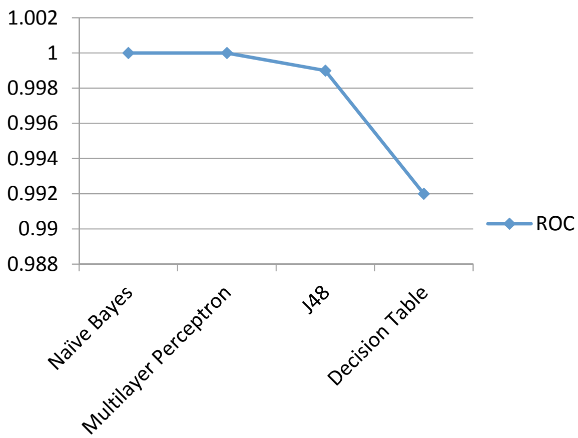


Fig. 8 ROC curve

selection based on chronic kidney disease dataset using genetic search algorithm is shown in Table 2.

The attributes selected by genetic search after 100 iterations are: 2, 3, 4, 6, 10, 12, 13, 14, 15, 16, 17, 19, 20, 22, 23, 24 = 16. That is, the selected attributes are {bp, sg, al, rbc, bgr, sc, sod, pot, hemo, pcv, wbcc, htn, dm, appet, pe, ane} from the dataset. The attributes which are reported by genetic search as irrelevant are neglected here. Taking relevant attributes selected by genetic search into consideration is the objective of our further study. That is, after attribute selection, we have applied all the six classification algorithms to see their behaviour for various parameters shown in Table 2.

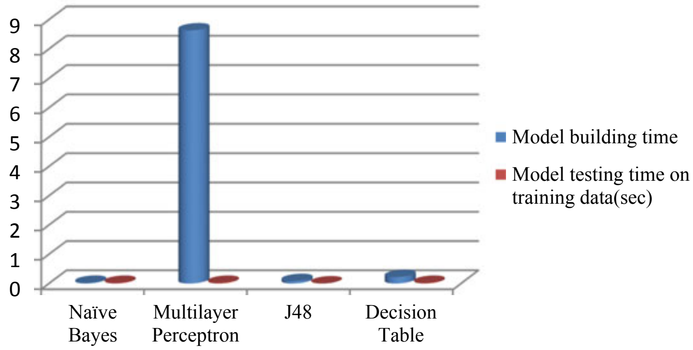


Fig. 9 Model building time and model testing time analysis

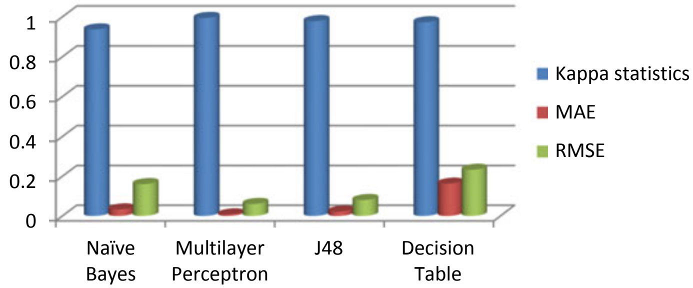


Fig. 10 Kappa statistics and error rate after feature selection analysis

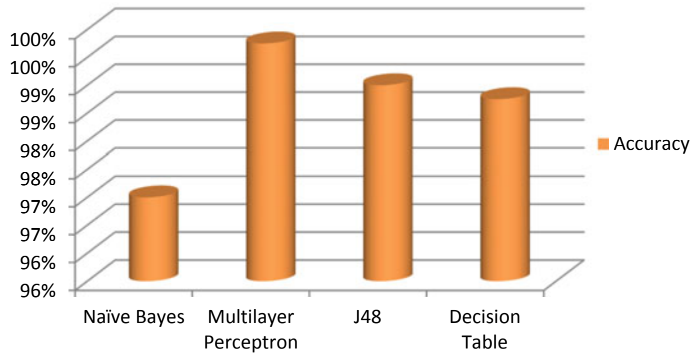


Fig. 11 Classification accuracy after feature selection

From Table 2 it is observed that the accuracy of Naïve Bayes classifier algorithm is increased by 2%, whereas, the accuracy of Decision Table is getting reduced mildly. Other algorithms have the same classification accuracy as shown previously

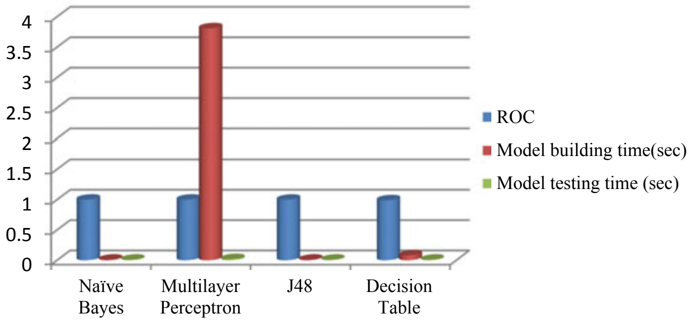


Fig. 12 ROC, model building time and model testing time after feature selection

in Table 1. From Table 2, it is interesting to note that values of other parameters for the classifier algorithms are also changed. Figures 10, 11, and 12 show the Kappa statistics and error rate, Accuracy, and ROC, Model building time and model testing time, respectively, after feature selection. From Fig. 10,11, and 12, it is observed that in all the cases Multilayer Perceptron's performance is quite better than all other classifiers being used for our experiment.

5 Conclusion

In this paper, the kidney disease is predicted using four algorithms namely Decision Table, J48, Multilayer Perceptron, and Naïve Bayes. A comparative study has been done between their classification results taking six parameters into consideration. The MLP classifier consistently outperformed other classifiers in terms of better accuracy and performance in risk prediction to predict kidney disease. Thus, it is concluded that Multilayer Perceptron's performance is quite better than all other classifiers being used for our experiment with the dataset in all instances.

References

1. Dhamodharan, S.: Liver disease prediction using Bayesian classification. In: Special Issues, 4th National Conference on Advance Computing, Application Technologies, May 2014
2. SolankiA, V.: Data mining techniques using WEKA classification for sickle cell disease. Int. J. Comput. Sci. Inf. Technol. **5**(4), 5857–5860 (2014)
3. Joshi, J., Rinal, D., Patel, J.: Diagnosis and prognosis of breast cancer using classification rules. Int. J. Eng. Res. Gen. Sci. **2**(6), 315–323 (2014)
4. David, S.K., Saeb, A.T., Al, Rubeaan K.: Comparative analysis of data mining tools and classification techniques using WEKA in medical bioinformatics. Comput. Eng. Intell. Syst. **4**(13), 28–38 (2013)

5. Vijayarani, S., Sudha, S.: Comparative analysis of classification function techniques for heart disease prediction. Int. J. Innov. Res. Comput. Commun. Eng. **1**(3), 735–741 (2013)
6. Kumar, M.N.: Alternating decision trees for early diagnosis of dengue fever. arXiv:1305.7331,2013
7. Durairaj, M., Ranjani, V.: Data mining applications in healthcare sector a study. Int. J. Sci. Technol. Res. IJSTR **2**(10) (2013)
8. Sugandhi, C., Ysodha, P., Kannan, M.: Analysis of a population of cataract patient database in WEKA tool. Int. J. Sci. Eng. Res. **2**(10) (2011)
9. Yasodha, P., Kannan, M.: Analysis of population of diabetic patient database in WEKA tool. Int. J. Sci. Eng. Res. **2**(5) (2011)
10. Bin Othman, M.F., Yau, T.M.S.: Comparison of different classification techniques using WEKA for breast cancer. In: 3rd Kuala Lumpur International Conference on Biomedical Engineering 2006, pp. 520–523. Springer, Berlin Heidelberg (2007)
11. Lambodar, J., Kamila, N.K.: Distributed data mining classification algorithms for prediction of chronic-kidney-disease. Int. J. Emerging Res. Manage. Technol. **4**(11), 110–118 (2015). ISSN 2278-9359
12. Lambodar, J., Ramakrushna, S.: Chronic disease risk prediction using distributed machine learning classifiers. In: Proceedings of IEEE International Conference on Information Technology (ICIT) 2017. © 2017 IEEE. https://doi.org/10.1109/icit.2017, 46978-1-5386-2924-6/17
13. Lambodar J., Kamila, N.K.: A model for prediction of human depression using apriori algorithm. In IEEE Xplore Digital Library Proceedings, pp 240–244 (2014). ISBN 978-1-4799-4152-0

A Review on Machine Learning Algorithms

Sushree Sasmita Dash, Subrat Kumar Nayak, and Debahuti Mishra

Abstract In today's world, machine learning is promising to produce consistently accurate estimates in different applications. The machine learning system learns effectively from the training dataset of different completed projects. It has been proved by researchers that machine learning algorithms can execute the entire work automatically, once it learns to handle the data. The paper presents a systematic review of several machine learning algorithms. The use of these machine learning algorithms can be found in many areas, such as image processing, predictive analytics, and data mining, etc. The intention of this review is to help other researchers to carry out their research on expert estimation techniques using various machine learning tools.

Keywords Machine learning · Supervised learning · Unsupervised learning · K-means · Reinforcement

1 Introduction

Machine Learning (ML) is the subset of Artificial Intelligence that provides computer systems the ability to simulate human intelligence. ML teaches machines to handle data efficiently. Mainly ML is used to interpret the pattern present in the data and to extract information from it [1]. It enables computer systems for searching and identifying hidden information or patterns, without being programmed explicitly, when exposed to new data sets. The demand for ML is increasing day by day because of the presence of an abundant amount of data all over the world. Nowadays, ML

S. S. Dash (✉) · S. K. Nayak · D. Mishra
Department of CSE, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: sasmitadash83@gmail.com

S. K. Nayak
e-mail: subratnayak@soa.ac.in

D. Mishra
e-mail: debahutimishra@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_51

Fig. 1 Types of learning [2, 3]

```
Tree_learning(T, Target, ar)
Steps:
Create a node
If T has same target attribute ti then return the single root node tree with target attribute ti
 if ar=null
then return the single root node tree having most common value of the target in T
else
Based on an entropy-based measurement select the attribute A from ar for the best classification of T
Set the attribute A for root node
Repeat for each legal value of A and vi,
do:
   add a branch having the value A= vi, below root node
   if Tvi= null
      then add a leaf node having most common value of the target in T below that branch
      else
   add the sub-tree below that branch which is learned by Tree_learning(Tvi, Target, ar{A})
   return root
```

is used to extract meaningful information in many industries, which ranges from medicines to military.

The goal of ML is to make machines learn from the examples. Many researchers have done tremendous work regarding how to make machines learn by themselves [2, 3]. Some of the several approaches of ML techniques are shown in Fig. 1.

The rest of the manuscript has been categorized into the following sections. Section 2 explains all of these ML techniques. Section 3 concludes this paper.

2 Learning Categories

2.1 Supervised Learning

Basically, a supervised learning algorithm needs a supervisor as a teacher. Here the input dataset, which is given to the system, is divided into two parts: training dataset and testing dataset. In this learning technique, the machine is trained or taught using some well-labeled data, otherwise called as the training dataset. From the training dataset, different algorithms of ML learn about the hidden patterns present in the labeled data and produce a correct prediction or classification to the testing dataset [4]. Here, three mostly used algorithms of supervised machine learning have been discussed.

1. **Decision Tree**. A decision tree has a structure like a flowchart in which internal node is represented as a test on an attribute, the leaf nodes represent the class labels and the branch is represented as conjunctions of attributes which belongs to those class label. The entire path from the root to the leaf represents the classification rule. The decision tree mainly groups the attributes into different classes by sorting the values associated with the attributes [4]. The diagram in Figure 2 illustrates the pseudocode for the decision tree in which T is the training set, Target is the target attribute and ar is the descriptive attribute. Figure 3 describes a decision tree example for taking the decision with labels (Raining (Yes), not raining (No)).

2. **Naïve Bayes**. These algorithms are probabilistic algorithms. These get the probability value by calculating the frequency and the combination of values from the related collections [5]. To do the classification using the Naïve Bayes algorithms, a set of data need to be provided. For that well-labeled data, Eq. 1 is applied.

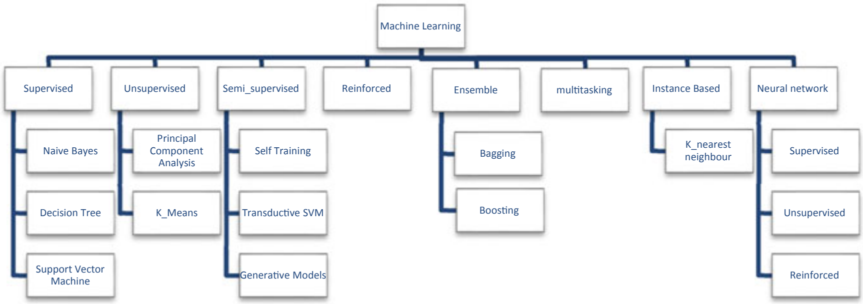
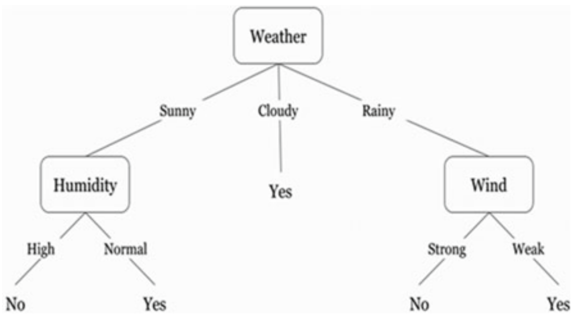


Fig. 2 Pseudo code for decision tree [15]

Fig. 3 Example of decision tree [16]



Input :
D= Training dataset
T= (t_1........t_n) // In testing dataset the value of predictor variable
Output : A class of testing dataset
Step 1: Read training dataset D;
Step 2: For each class calculate the standard deviation and mean of the predictor variables;
Step 3: Repeat step 4 until the probability of all the predictor variables (t_1........t_n) has been calculated;
Step 4: Calculate the probability for t_i in each class using gauss density equation;
Step 5: Calculate the likelihood for each class;
Step 6: Find the highest likelihood;

Fig. 4 Pseudo code for Naïve Bayes algorithm [17]

$$P(C|F_1 \ldots F_n) = \frac{P(C).P(F_1 \ldots F_n|C)}{P(F_1 \ldots F_n)} \quad (1)$$

In Eq. (1) C represents the class label and $F_1 \ldots F_n$ represents the characteristics required for classification. The example of Bayesian is given in Fig. 4 and the algorithm for the Naïve Bayes can be found in Fig. 5.

3. **Support Vector Machine (SVM)**. SVM is one of the most popularly applied ML techniques for classification. SVM follows the margin calculation principle. According to this algorithm, the classes are divided based on a straight line so that maximum distance is maintained between the line and the classes. This helps in minimizing the error of classification. The working of SVM is shown in Fig. 6 and the pseudocode in Fig. 7.

2.2 Unsupervised Learning

Unlike supervised learning algorithms, these algorithms use the information, which is not labeled or classified in training data. Without any guidance, the algorithms are able to group unsorted information considering the similarities and differences present between those data. Unsupervised learning is mainly used in clustering and feature reduction purposes [6]. Two of the popularly used unsupervised algorithms for application in clustering and dimensionality reduction have been discussed here.

1. **K-Means Algorithm**. K-means is a popular name in the field of clustering, an unsupervised machine learning approach. It makes inferences from the given datasets using input vectors without any known or labeled data. The objective of this algorithm is to cluster data points that are similar in nature and detect

Fig. 5 Pseudo code for SVM [18]

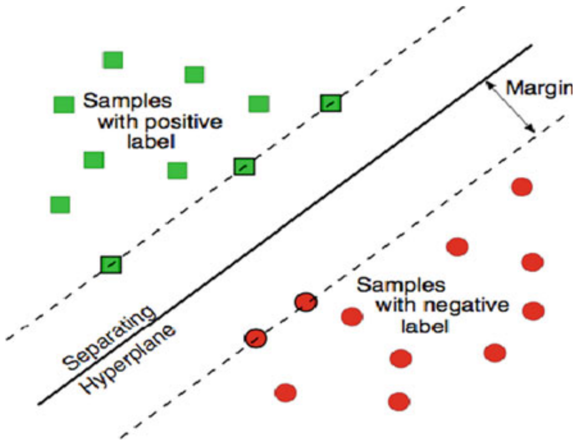
Input:
- n_{iv} : no. of vectors (input)
- n_{sv} : no. of vectors (support)
- n_{fe} : no. of features (support)

$VS[n_{sv}]$: array of support vector
$VI[n_{iv}]$: array of input vector
b^* : bias value
Output:
- F (decision function)

For $i = 1,2,\ldots,n_{iv}$
$F = 0$
For $j = 1,2,\ldots,n_{sv}$
- distance=0
- For k=1,2,...... n_{fe}
  - distance + = $(VS[j].\text{feature}[k] - VI[i].\text{feature}[k])^2$
- End
- $k = \exp(-y.\text{distance})$
- $F+ = VS[j].\alpha^* .k$

End
$F + = b^*$
End

Fig. 6 Working of support vector machine [18]



the underlying patterns. For this objective to be achieved, K-means clusters the dataset into a fixed (K) number of clusters. The algorithm finds the centroid of the cluster by finding the mean of the dataset. At first, the algorithm finds K number of centroids and then the data point is allocated to the nearest cluster by calculating the distance between the centroid and the data point. During this process, the algorithm tries to keep the size of the cluster as small as possible.

Fig. 7 K-means algorithm [19]

Input :
K= number of clusters
R= sets of lift ratio
Output:
Sets of K clusters
Procedure:
Randomly choose K objects from R as cluster centers
Repeat the following steps until no change:
Assign the objects to the most similar cluster by finding the mean of the object with in that cluster
Calculate the mean of the object for the assigned cluster and update cluster mean

Figure 8 shows clustered data and Fig. 10 represents the k-means algorithm (Fig. 9).

2. **Principal Component Analysis (PCA)**. PCA is a common form of factor analysis. It comes in the category of multivariate statistical techniques. The interrelationships between a large number of variables can be analyzed by using PCA. The PCA provides dimension reduction for faster and easier computation. It reduces the number of variables and clusters them into a manageable group called components or factors. The interrelated variables belong to one component. The key dimension is selected from these components. These key dimensions are used to cluster the population into homogeneous groups. An example of PCA is shown in Fig. 12. In which a 2D data is taken and plotted in the graph. It is taken up two axes before the Principal Component Analysis applied to it. After applying PCA

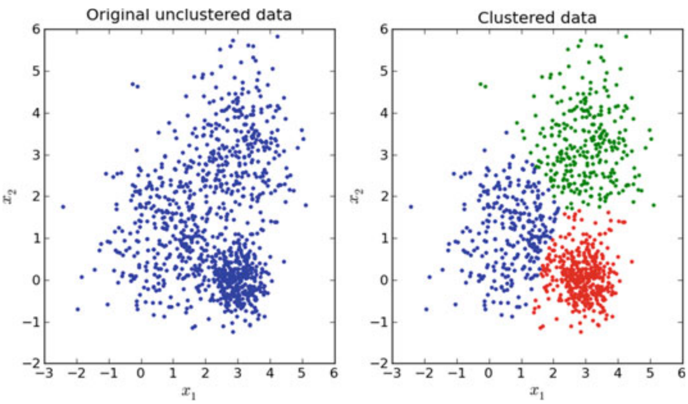


Fig. 8 Example of K-means clustering [7]

Fig. 9 Pseudo code for PCA [20]

$R <- X$

Repeat for $(k = 0,1,2,......,k-1)$ do

{ $\lambda = 0$

$T^k <- R^k$

for $(j = 0,....,j)$ do

{ $P^{(k)} <- R^T T^{(k)}$

$P^{(k)} <- P^{(k)} \| P^{(k)} \|^{-1}$

$T^{(k)} <- RP^{(k)}$

$\lambda' <- \| T^{(k)} \|$

if $(|\lambda' - \lambda| \leq \varepsilon)$ then break

$\lambda <- \lambda'$ }

$R <- R - T^{(k)} (P^{(k)})^T$ }

return T,P,R

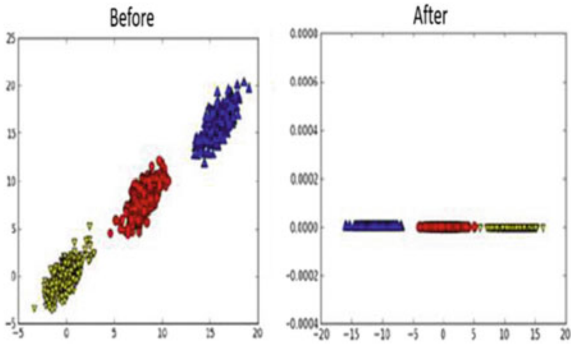


Fig. 10 The data before and after applying PCA [21]

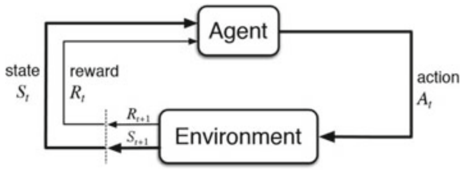


Fig. 11 Model of reinforcement learning [22]

on those data, it becomes 1D. Figure 9. Shows the data before and after applying PCA and Fig. 11 describes the pseudocode for Principal Component Analysis.

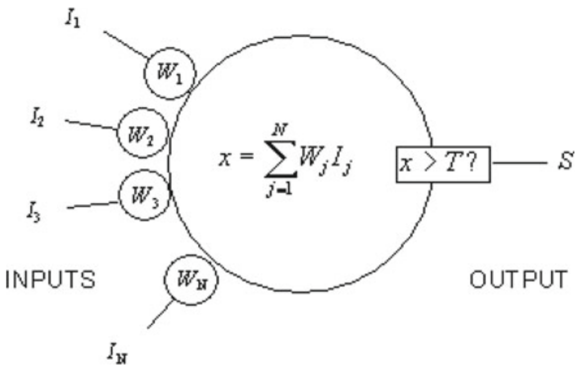


Fig. 12 An artificial neuron [23]

2.3 *Reinforcement Learning*

This is a type of machine learning, which trains the algorithm using a reward and punishment system. In this algorithm, the agent learns from the environment. When an agent performs correctly, it gets rewards, and when performs incorrectly gets a penalty. The agent learning is done by maximizing the reward and minimizing the penalty [7]. This is done by estimating the value of a particular state. This estimated value is then corrected over time by propagating part of the next state's reward. After the trial of all the states and actions for a sufficient amount of times, an optimal policy can be defined. Generally, the action which maximizes the value of the next state is picked. In Fig. 12, the learner receives the current state as input at time *t,* i.e., S_t and the reward from the environment, i.e., R_t. Based on these values, an action A_t is taken by the agent which generates a new state S_{t+1} and a reward R_{t+1}.

2.4 *Multitask Learning*

Multitask learning is used to make the performance better for other learners. When this algorithm is applied to a particular task the whole procedure for solving the problem is remembered by it. Thereafter, it solves other similar types of problems by using these steps. Here, one algorithm helps another algorithm and this mechanism is known as an inductive transfer mechanism [8]. Because of the sharing of experience the algorithms can learn simultaneously instead of individually. Therefore, the learning process is also faster.

2.5 Ensemble Learning

In an ensemble learning method, multiple learners such as classifiers or experts solve a particular problem together [9]. Different algorithms like the decision tree, Naïve Bayes or neural network, etc., can be used as individual learners. Since 1990 ensemble learning becomes a hot research area. Many observations regarding ensemble learning prove that a number of learners are always better than an individual learner for doing a particular job. Here, two mainly used ensemble learning algorithms are discussed [10].

1. **Boosting**. Boosting: Boosting is used to decrease variance and bias. The main objective of this algorithm is to create a number of learners, which are weak in nature but gradually make them stronger. Strong learners or algorithms are those having a strong correlation with true classification but weak learners are those which have a poor correlation with the true classification [10].
2. **Bagging**. Bagging is also known as bootstrap aggregating. It is used to increase the accuracy and stability of the learning algorithm. Mainly it is applied in regression and classification. It also reduces variance, which in turn helps to handle overfitting [11].

2.6 Semi-supervised Learning

The concepts of semi-supervised learning come to overcome these disadvantages of supervised and unsupervised learning [12]. Here, the algorithm works upon both unlabeled and few labeled data. At first, the algorithm clusters similar types of data using the unsupervised learning algorithm, and then the rest will be labeled by taking the help of previously taken a few numbers of labeled data. Three of the several methods of semi-supervised learning are detailed as follows [13].

1. **Generative Models**. These models are the simplest method of semi-supervised learning. It first estimates the mixed probability $P(q, r)$. It can be calculated as follows:

$$P(q, r) = P(r).P(q|r) \tag{2}$$

   where $p(q|r)$ is the mixed distribution of data points, that belong to each class [7]. For each component, only one example of labeled data is enough to make the mixture distribution.
2. **Self-training**. Here, the classifier tries to train itself. Hence, the process is called as self-training. In this method, a portion of labeled data is used to train the classifier first. Thereafter, it operates on unlabeled data. Then the label, which is predefined, is added with the unlabeled points and this procedure is repeated many times.

3. **Transductive Support Vector Machine**. Support Vector Machine is extended to build a transductive support vector machine. It considers both labeled and unlabeled data. Within the labeled and unlabeled data, the margin is maximum here. The TSVM comes in the class of a nondeterministic polynomially hard problem.

2.7 Neural Network

Artificial neural network (ANN) is a system of information processing that is inspired by the human brain's neural network. The neuron present in a human brain is mathematically represented in ANN. A biological neuron mainly has four parts: nucleus, dendrites, axon, and synapse. Dendrites are the input unit of a neuron, which receives electrical signals. Nucleus processes it, and axon acts as the output terminal, which carries the processed signal to the dendrites of another neuron [14]. Synapses are the endpoints of axons that allow the signal transfer from one neuron to another. Its size changes in response to learning. The interconnection between more than one neuron is known as a neural network.

1. **Supervised Neural Network**. In this network, the result of the algorithm for a particular set of data is known previously. This output is compared with the calculated output of the neural network and an error value is calculated. Then parameters, i.e., weights and bias value are changed, based on that error. This process is done several times for the learning of the neural network. The main strategy used in a feedforward neural network is the supervised neural network (Fig. 13).
2. **Unsupervised Neural Network**. In this type of algorithm, the output is previously unknown for any input. Its main objective is to cluster the data according to some similar characteristics present between them. It is done by checking the correlation present between different inputs and grouping them accordingly (Fig. 14).
3. **Reinforced Neural Network**. Here, the network learns from the environment by communicating with it. Feedback is given to the neural network from the environment about the decision of the network whether it is correct or incorrect. If the decision is correct, it strengthens the connection related to the particular output. Otherwise, the connection becomes weaker (Fig. 15).

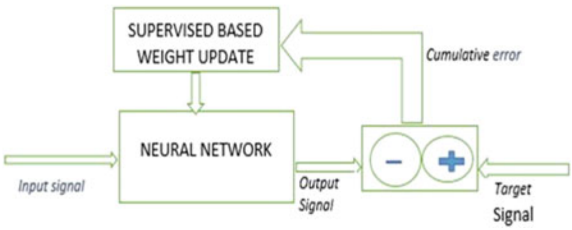


Fig. 13 Supervised neural network [24]

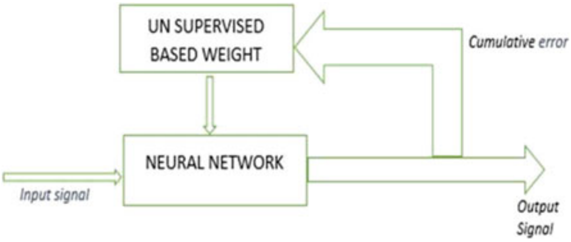


Fig. 14 Unsupervised neural network [24]

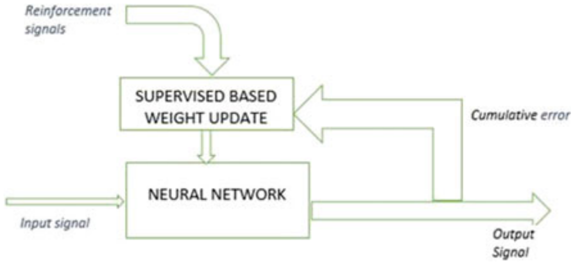


Fig. 15 Reinforced neural network [24]

2.8 Instance-Based Learning

In this learning, the particular pattern type is being learned by the learner. The pattern has applied to new data. So it is known as Instance-based. The learner waits for the data, once data is given then it acts on the training data and stores it in memory. When testing data is given to it, rather making a generalized solution, it compares that particular instance with the instances which are already stored in the memory. The size of the data affects the complexity of the algorithm. The example of this type of learning is explained below.

1. **K-Nearest Neighbor (KNN)**. In KNN, the learner fed the training dataset. The test data is introduced and compared with training data. k number of data that are most correlated to each other is taken from the training set. The new class for the test data takes the majority of k [7]. The KNN algorithm is given in Fig. 16.

3 Conclusion

The aim of this review is to discuss different machine learning algorithms. Machine learning is used everywhere knowingly or unknowingly nowadays, such as getting product lists in online shopping to digital marketing, etc. After analyzing all the techniques, we can state that each technique can be used in different application areas, and based on the advantages of each algorithm; it is useful in different domains. We can choose the best technique by keeping in mind the limitations and the prime

Fig. 16 Pseudo code for KNN [25]

k-Nearest Neighbor

Classification (T, L, s) $//T$: training data, L : class labels of T , s : unknown sample data

For $i = 1,2,3.....m$ do

Compute distance (T_i, s)

end for

Compute set I containing indices for the k smallest distances (T_i, s)

return majority label for $\{L_i \, where i \in I\}$

focus in the efficiency and performance of each of the techniques for a particular application.

References

1. Richert, W., Coelho, L.P.: Building Machine Learning Systems with Python. Packt Publishing Ltd. (2013). ISBN 978-1-78216-140-0
2. Welling, M.: A First Encounter with Machine Learning, vol. 12. University of California, Irvine, CA (2011)
3. Bowles, M.: Machine Learning in Python: Essential Techniques for Predictive Analysis. Wiley (2015)
4. Kotsiantis, S.B., Zaharakis, I., Pintelas, P.: Supervised machine learning: A review of classification techniques. Emerging Artif. Intell. Appl. Comp. Eng. **160**, 3–24 (2007)
5. Lowd, D., Domingos, P.: Naive Bayes models for probability estimation. In: Proceedings of the 22nd International Conference on Machine Learning. ACM (2005)
6. Shalev-Shwartz, S., et al.: PEGASOS: primal estimated sub-gradient solver for Svm. Math. Program. **127**(1), 3–30 (2011)
7. Dey, A.: Machine learning algorithms: a review. Int. J. Comput. Sci. Inf. Technol. **7**(3), 1174–1179 (2016)
8. Caruana, R.: Multitask learning. Mach. Learn. **28**(1), 41–75.D (1997)
9. Opitz, D., Maclin, R.: Popular ensemble methods: an empirical study. J. Artif. Intell. Res. **11**, 169–198 (1999)
10. Zhang, M.-L., Zhou, Z.-H.: Exploiting unlabeled data to enhance ensemble diversity. Data Min. Knowl. Disc. **26**(1), 98–129 (2013)
11. Kim, D.G., et al.: Developing of new a tensorflow tutorial model on machine learning: focusing on the Kaggle titanic dataset. IEMEK J. Embedded Syst. Appl. **14**(4), 207–218 (2019)
12. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. Synth. Lect. Artif. Intell. Mach. Learn. **3**(1), 1–130 (2009)
13. Zhu, X.J.: Semi-supervised Learning Literature Survey. Department of Computer Sciences, University of Wisconsin-Madison 2005
14. Sharma, V., Rai, S., Dev, A.: A comprehensive study of artificial neural networks. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **2**(10) (2012). ISSN 2277128X
15. Hu, Y.-J., et al.: Decision tree-based learning to predict patient controlled analgesia consumption and readjustment. BMC Med. Inf. Decis. Making **12**(1), 131 (2012)

16. Neves, T.FM.: A data mining approach to predict probabilities of football matches (2019)
17. Saputra, M.F.A., Widiyaningtyas, T., Wibawa, A.P.: Illiteracy classification using K means—Naïve Bayes algorithm. JOIV: Int. J. Inf. Vis. **2**(3), 153–158 (2018)
18. Lim, C., Lee, S.-R., Chang, J.-H.: Efficient implementation of an SVM-based speech/music classifier by enhancing temporal locality in support vector references. IEEE Trans. Consum. Electron. **58**(3), 898–904 (2012)
19. Zhou, P.-Y., Chan, K.C.C.: A model-based multivariate time series clustering algorithm. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham (2014)
20. Andrecut, M.: Parallel GPU implementation of iterative PCA algorithms. J. Comput. Biol. **16**(11), 1593–1599 (2009)
21. Harrington, P. Machine Learning in Action. Manning Publications Co. (2012)
22. 5 Things you need to know about reinforcement learning. https://www.kdnuggets.com/2018/03/5-things-reinforcement-learning.html. Accessed 12 Oct 2019
23. Crash Introduction to Artificial Neural Network. https://ulcar.uml.edu/~iag/CS/Intro-to-ANN.html. Accessed 14 Oct 2019
24. Hiregoudar, S.B., Manjunath, K., Patil, K.S.: A survey: research summary on neural networks. Int. J. Res. Eng. Technol. **3**(15), 385–389 (2014)
25. Tay, B., Hyun, J.K., Oh, S.: A machine learning approach for specification of spinal cord injuries using fractional anisotropy values obtained from diffusion tensor images. Comput. Math. Methods Med. (2014)

Bigdata Analytics and Deep Learning in Early Detection of Neurogenerative Diseases

Padmini Mansingh, Binod Kumar Pattanayak, and Bibudhendu Pati

Abstract The emergence of big data analytics is an extremely powerful tool for assessment of the huge amount of data gathered from different processes in enterprises and industries. It has been extremely popular across researchers in various fields, scientists and industry personnel as well. These days, big data analytics has been the most useful means for assessment of medical data obtained from patients and decision making for the facilitation of the desired treatment of the patient. Today, big data analytics is a popular term in the healthcare industry. A continuum of ill-health, patients are now involving their life for a better understanding of the diseases to get an effective result through different new tools of big data analytics. So, now researchers begin to build effective models by testing large/high voluminous data. In this paper, we address clinical support for neurogenerative diseases by merging two high-end techniques, i.e., deep learning and big data analytics.

Keywords Big data · Health care · Alzheimer's diseases · Parkinson's diseases · Huntington's diseases · Machine learning · Deep learning

P. Mansingh (✉) · B. K. Pattanayak
Department of Computer Science and Engineering, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha, India
e-mail: padminimansingh@soa.ac.in

B. K. Pattanayak
e-mail: binodpattanayak@soa.ac.in

B. Pati
Department of Computer Science, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha, India
e-mail: patibibudhendu@gmail.com

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

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_52

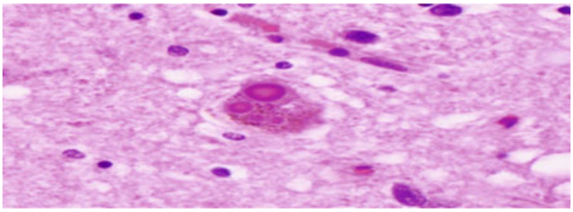


Fig. 1 Neurodegenerative disease: Lewy bodies commonly found in pigmented neurons present in Parkinson's disease

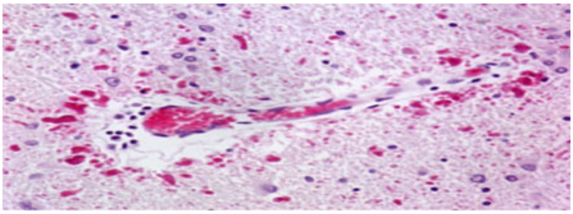


Fig. 2 Development of neurodegenerative: image: tissue stains from an eight-year-old Alexander disease patient

1 Introduction

In recent years, researchers noticed the way to provide a relevant solution by analyzing and improving medical equipment for immediate detection of neurogenerative diseases. The challenges which are generated by biomedical researchers are the most important concepts for regenerative medicine or disease remission. A significant barrier is generated when current clinical practices go through the tunnels of big data analytics. So, the development of new technologies can overcome the difficulties, which are generated by traditional clinical equipment. For medical support, deep learning technique is introduced to combine with big data analytics for better result. For patient diagnosis, different software and methods are developed and shared by researchers for further study of neurodegenerative diseases with predictive big data analytics applications [1]. Some of the histopathological images listed below of neurogenerative diseases are shown in Figs. 1 and 2 [2, 3].

2 Motivation

Application of deep learning in health care in combination with big data analytics can be able to yield better results by virtue of using more sophisticated algorithms for retrieval of essential features of voluminous image data, and this is the principal intention behind this current research work. Deep learning is one of the important fields for computer vision that makes use of massive training artificial neural network (MTANN) and convolutional neural network (CNN) for achieving a higher performance [4]. End-to-end deep learning techniques tend to solve the problems of different neurogenerative diseases for getting an effective result.

Multifactorial debilitating disorders of the nervous system are directly related to neurogenerative diseases that are rapidly generated as a consequence of misfolding and dysfunctional trafficking of proteins.

Every year, above millions of individuals worldwide are directed toward the treatment of neurodegenerative diseases and neuroscientists are fully dedicated to recently innovated different treatment plans for obtaining a proper diagnosis procedure [5]. Odor identification, recognition and detection of the threshold are the three olfactory domains that are qualified to control both Alzheimer's disease (AD)- and Parkinson's disease (PD)-affected patients [6].

3 Predictive Analytics of Neurogenerative Diseases

Neurons are the main components of the human nervous system. A neuron cannot rebuild itself when it becomes damaged or dies, rather it causes different mental disorders. It affects millions of old-age people worldwide. The burden of dementia exceeds day by day due to the progressive nature of discerning disintegration of specific neuronal populations. Different neurodegenerative diseases include spinocerebellar ataxia (SCA), spinal muscular atrophy (SMA), Prion disease, Huntington's disease (HD), Alzheimer's disease (AD) dementia, Parkinson's disease (PD), PD-related disorders and other motor neuron diseases (MND) as shown in Fig. 3.

Neurodegenerative disease is a central point for all kinds of diseases in the human body. Death of the nervous system causes these types of neurodegenerative diseases and is incurable. In recent times, the mechanism study of heritable changes in gene expression or phenotype has emerged as a global burden of dementia researchers [7]. Neurodegenerative diseases generally affect memory or movement of different

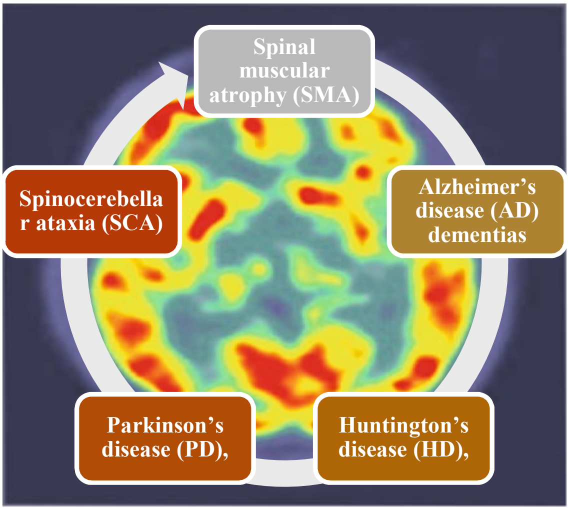


Fig. 3 Different types neurogenerative diseases

parts of the human body. In recent years, the essential features of each disease are briefly described through the current status of deep learning with a huge amount of big clinical data for better classification of stages of neurogenerative diseases. The different limitations of genetic mouse models for thoughtful main neurodegenerative diseases including Alzheimer's, Parkinson's and Huntington's disease and amyotrophic lateral sclerosis are described by researchers in [8]. One among five Americans is notoriously difficult to be treated for dementia. Thanks to big data analytics, researchers categorized the disease with respect to risk factors. Due to the inability of traditional methods, it is impossible to cope with complex analyzation of a positive outcome. By finding the hidden pattern, more medical records are digitized for symptoms suggestive of dementia, thereby prompting diagnostic tests, analysis of a vast range of variables that lead to the exposure toward more effective treatments. However, it generates a vast range of analyzing data for enabling us to develop to detect early target dead neurons. So, each patient treated specially by the related doctor by virtue of comparing various treatment plans since one treatment plan may work better than others in order to cure diseases by changing their life cycles through analysis of petabytes of data for saving a life. Millions of patients have been treated for dementia on the basis of the changes in behavioral patterns with respect to time [9]. The neurogenerative disease-affected patients can be monitored for references to an illness. Ground-breaking achievements in tracking dead neurons contributed to the maximum for tackling diseases. However, a huge amount of data from social media has been successfully used to map and has even helped to predict the disease at its early stage. Deep learning models have been significantly useful for experts in achieving better investigation results.

4 Deep Learning Approaches

Remarkable usage of deep learning in healthcare systems opens huge opportunities that may lead to significant benefits to the patients. Deep learning or deep neural learning (DNN) is a part of machine learning which is capable of learning the unstructured data. The extraction of valuable features from the unstructured vast amount of data is beyond the capacity of traditional methods. So, deep learning comes with unbelievable potential with self-adaptive algorithms for a better analysis. Deep learning approaches are used in different complex tasks such as speech recognition, image interpolation and language translation. Through deep learning, researchers precisely localized the objects of various classes of diseases at a low cost.

4.1 Limitations of Traditional Machine Learning

One of the challenges in traditional machine learning models is feature extraction. The programmer needs to be specific for critical problems like object recognition and handwriting recognition. Deep neural networks became popular as one of the few methods that can help in automatic feature extraction. If there are many hidden layers in between input and output layers, then it becomes a deep neural network that can be used to minimize cost, apart from the stochastic gradient descent approach as an optimization method. Now, there are options like AdaGrad, Adam optimizer for massive computational operations. We mostly use the gradient descent approach as the deep neural networks (DNNs) model for complex nonlinear relationships. The main purpose of deep learning (DL) models is that it produces much better results than normal ML networks.

4.2 Convolutional Deep Neural Networks

If we increase the number of hidden layers, the number of weights and biases will also increase. So, CNN is one of the best solutions for removing the complexity generated by traditional machine learning models. CNN is broadly used in computer vision; the inkling behind convolutional neural networks is the concept of a "moving filter" which passes through the image. CNN drastically reduces the number of parameters that need to be changed. Big data and cloud computing are the two most important trends used for decision making as well as new emerging analytical techniques for extracting a large amount of valuable information from the healthcare industry.

4.3 Challenges in Deep Learning Algorithms

Overfitting and computation time are two main challenges in deep learning which need to be overcome because of added layers of abstraction. Fighting for overfitting is managed by different methods of regularization.

4.3.1 Data Augmentation

New data are augmented using existing data by applying some transformations on it, and the transformations totally depend on its architecture. Computer vision task such as object classification represents an effective data augmentation technique that relies on adding new data points that are cropped or translated versions of original data, and the resulting augmented dataset may appear to be many times the size of the original dataset. By virtue of creating the modified dataset from actual data after resizing is a well-known technique of augmentation.

4.3.2 Transfer Learning

A pre-trained model acts as a feature extractor and "fine-tuning" the model with our own dataset is called transfer learning. For taking a better decision, a set of algorithms required in machine learning is called deep learning. A quickly adoptable promising technology creates powerful deep learning tools. The fully popular open-source list of tools is shown in Table 1.

Mathematical notation:

Source: domain = Q_s; learning task = P_s; target: domain = Q_t; learning task = P_t; Feature space = U; label space = W. By gaining information from Q_s and P_s, we improve our learning rate of conditional probability $P\ (W_t|U_t)$ in the target domain. For minimizing the gap of transfer learning is possible only when $Q_t \neq Q_s$ or $P_t \neq P_s$. Single source is considered only for a single source domain. According to the theorem of both domain and task, we obtain either $Q_t \neq Q_s$ or $P_t \neq P_s$. Here, $U_t = U_s$ and $W_t = W_s$ is specified for homogenous transfer learning. Elimination of the space between the data distributions of source and target domains can be possible by addressing the $P(U_t) \neq P(U_s)$ and/or $P(W_t|U_t) \neq P(W_s|U_s)$ [10]. The solutions for homogeneous transfer learning problems are being conducted in the following steps:

1. Target eliminating marginal distribution: $(P(U_t) \neq P(U_s))$;
2. Correct for the conditional distribution difference: $(P(W_t|U_t) \neq P(W_s|U_s))$;

The solution for heterogeneous transfer learning of $U_t \neq U_s$ and $W_t \neq W_s$ is possible by removing the gap between reducing the problem and feature space [11].

Table 1 Different deep learning tools

Tools	Developed by	Year	License	Core language
Keras	François Chollet	2015	MIT	PYTHON
Pytourch	Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan	2016	BSD	Python, C++, CUDA
Caffe	Yangqing Jia	2017	BSD	C++
Theano	Montreal Institute for Learning Algorithms (MILA)	2017	BSD	Python, CUDA
TensorFlow	Google Brain Team	2015	Apache License 2.0	Python, C++, CUDA
MXNet	Apache Software Foundation	2016	Apache License 2.0	C++, Python, R, Julia, JavaScript, Scala, Go, Perl
Deeplearning 4j	Alex D. Black, Adam Gibson, Josh Patterson	2017	Apache License 2.0	Java, Scala, CUDA, C, C++, Python, Clojure
Chainer	Seiya Tokui	2015	MIT	Python

4.3.3 Early Stopping

Using the gradient descent approach, we train our neural network model. We stop training when the error starts increasing that is the main aim behind early stopping technique [12]. It is a kind of regularization technique used to avoid overfitting. The idea behind early stopping is relatively simple:

Step 1: Divide the data into two categories training and test sets;
Step 2: For every N epoch:

Calculate the performance of the network on the test set;
if the calculation result of the current model is superior to the previous model, then save the network;

4.3.4 Dropout

Dropout is a very well-known regularization technique for neural networks that are patented by Google. In this technique, for each iteration of gradient descent, we drop a set of aimlessly particular nodes and each neuron is kept with a probability of y or dropped randomly with probability $(1 - y)$. During evaluation and prediction, no dropout is used. The output of each neuron is multiplied by y so that the input to the next layer has the same expected value.

5 Conclusion

There is a vast amount of medical data for future analysis in order to overcome several issues regarding complexity of healthcare industry. Today, deep learning facilitates different tools for effective analysis of diagnosis, classification, survival risk of neurogenerative diseases. However, to effectively use deep learning tools in health care by combining different types of laboratory findings, demographic data, doctors' free-text and imaging data, neuroscientists can be able to make better models with data privacy and security. This paper can facilitate a new framework for eliminating some of the challenges of big data with deep learning in health care to prevent diseases, infections and illnesses. Here, we have highlighted most of the technologies of deep learning with respect to effective diagnosis required for medical images to be closely analyzed. Hence, deep learning algorithms are better to over limitation of existing traditional machine learning approaches which take much more time for identifying malicious spots of various diseases.

References

1. Dinov, I.D., Heavner, B., Tang, M., Glusman, G., Chard, K., Darcy, M., Madduri, R., Pa, J., Spino, C., Kesselman, C. Foster, I: Predictive big data analytics: a study of Parkinson's disease using large, complex, heterogeneous, incongruent, multi-source and incomplete observations. PloS one **11**(8): e0157077
2. Neuropathology: LewyBody (2019). https://adventuresinneuropathology.com/2018/08/16/lewy-body/. Accessed 10 Oct 2019
3. LabmedicaInternationalstaffwriters: Development of neurodegenerative diseases linked to astrocyte nitric oxide signaling (2015). https://www.labmedica.com/bioresearch/articles/294761792/development-of-neurodegenerative-diseases-linked-to-astrocyte-nitric-oxide-signaling.html. Accessed 10 Oct 2019
4. Hinton, G.: Deep learning—a technology with the potential to transform health care. Jama 320(11):1101–1102 (2018)
5. Sheikh, S., Haque, E., Mir, S.S.: Neurodegenerative diseases: multifactorial conformational diseases and their therapeutic interventions. J. Neurodegener. Dis. 1–8 (2013)
6. Mesholam, R.I., Moberg, P.J., Mahr, R., Doty, R.L.: Olfaction in neurodegenerative disease: a meta-analysis of olfactory functioning in Alzheimer's and Parkinson's diseases. Arch. Neurol. **55**(1), 84–90 (1998)
7. Lardenoije, R., Pishva, E., Lunnon, K, van den Hove, D.L.: Neuroepigenetics of aging and age-related neurodegenerative disorders, vol. 158. In: Progress in Molecular Biology and Translational Science, pp. 49–82. Academic Press (2018)
8. Trancikova, A., Ramonet, D, Moore, D.J.: Genetic mouse models of neurodegenerative diseases. In: Progress in Molecular Biology and Translational Science, pp. 419–482, vol. 100. Academic Press (2011)
9. Pal, R., Larsen, J.P., Moller, S.G.: The potential of proteomics in understanding neurodegeneration. Int. Rev. Neurobiol. **121**, 25–58 (2015)
10. Weiss, K., Khoshgoftaar, T.M., Wang, D.A.: Survey of transfer learning. J. Big Data **3**(1), 9 (2016)
11. Asgarian, A.: An Introduction to Transfer Learning (2018). https://medium.com/georgian-impact-blog/transfer-learning-part-1-ed0c174ad6e7. Accessed 14 Oct 2019
12. DL4J: What is early stopping? (2018). https://deeplearning4j.org/docs/latest/deeplearning4j-nn-early-stopping. Accessed 14 Oct 2019

Optimization Through Cuckoo Search with a Brief Review

Mitali Madhusmita Sahoo, Rasmita Dash, Rasmita Rautray, and Rajashree Dash

Abstract Optimization is a process in which the system is being modified so that the features can work more effectively or by finding any necessity execution. This process is conducted through maximizing and minimizing the parameters that are being involved in the problem. Many successful optimization approaches are available in the recent scenario; however, this research work focuses on cuckoo search technique. A new higher level procedure optimization algorithm called cuckoo search (CS) was developed by Yang and Deb in 2009. Cuckoo search is a swarm intelligence (SI) algorithm. Swarm intelligence techniques are those techniques that are being originated from the behavior of swarm. Cuckoo search algorithm is being inspired by the bird called cuckoo. It is the inspiration of brooding parasitism of cuckoo species and implemented successfully in many areas for optimization. This paper represents a brief review on CS algorithm along with the optimization process implementing some standard test functions. This research work highlights test function optimization for the non constraint function. This work may give an inspiration to apply the CS algorithm to solve the engineering design optimization problems.

Keywords Swarm intelligence algorithms · Cuckoo search algorithms · Non constraint function optimization

---

M. M. Sahoo · R. Dash (✉) · R. Rautray · R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha 751030, India
e-mail: rasmitadash@soa.ac.in

M. M. Sahoo
e-mail: mitalimadhusmita1322@gmail.com

R. Rautray
e-mail: rashmitaroutray@soa.ac.in

R. Dash
e-mail: rajashreedash@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_53

1 Introduction

Swarm intelligence is a way of behaving that shows a willingness to obey rules and deals with natural and artificial systems using self-organization. This communication frequently derives from the nature, especially from biological systems [1]. Many human artifacts drop into the state of swarm intelligence; some of the multi-robots systems as well as many computers programs have been systematically collected to interchange optimization problem based on data analysis. Some of the examples learned by swarm intelligence are ants and terminates colonies, fish schooling, crowds of birds, flocks of land animals and microbial intelligence.

Few representatives of swarm optimization algorithms are particle warm optimization (PSO), ant colony optimization (ACO), BAT algorithm, firefly algorithm, bee algorithm, artificial bee colony (ABC), etc. Short descriptions of each category are as follows: Particle swarm optimization (PSO) algorithms are nature-inspired population-based metaheuristic algorithms. It is a method of computation that solves a problem by repetition seeking for the improvement of an applicant solution accompanied by the esteemed to a given measurement of the quality. These algorithms imitate the collective behavior of birds flocking and fish schooling [1]. Ant colony optimization (ACO) is an ant behavior study-based method that can be used to find solution to optimization problem [2]. BAT algorithm is a metaheuristic algorithm used for global optimization. This algorithm is been inspired by the bio-sonar behavior of bats with degree of pulse rate and capable of noise [3]. Main purpose for a firefly flash is that it passes the signal to attract other fireflies. The movement of fireflies is random when their brightness becomes same [3]. The bee's algorithm is a population-based search algorithm. Bee algorithm often mimics for the food by bearing the search of honey bee colonies. Artificial bee colony (ABC) algorithm imitates the appearance of honey bees that search widely for food and applied successfully to different problems. This algorithm is a member of swarm intelligence algorithms. This algorithm is being proposed by Karaboga [4].

Inspired from the successful application of many soft computing approaches, this research work takes care of cuckoo search technique. As it does not deal with many parameter optimization, this algorithm is simple to implement and does not lie at local optima. The basic motivation of this paper is cuckoo search implementation for unconstraint function optimization and its application in various real-world scenarios.

The organization of this script is as follows. Section 2 highlights the characteristics of CS through its algorithm and Levy flight operation. Broader application of CS algorithm is shown in Sect. 3. Thereafter, the implementation of CS is highlighted in Sect. 4. The concluding remark and future work is presented in Sect. 5.

2 Basic of Cuckoo Search

Cuckoo is one of the interesting birds, not only for their beautiful sounds but also for its trending reproduction procedure. Some cuckoo lay their eggs in nest called communal nest so that it can remove other eggs to increase its hatching probability of its own egg. Communal nest is the nest where it is being shared by different birds. Some cuckoos are expert in mimicry process. The cuckoo mimicries the eggs of the birds in such a way that its been difficult for the host bird to find out its own egg. It gains the probability of survival of the cuckoo egg. The cuckoo egg hatching procedure is slightly faster than the other birds. Once the cuckoo bird comes out of the eggs, its first step is to kill the other bird's egg by pushing outside the nest. So that it can survive.

2.1 Representation of Cuckoo Search

Each cuckoo can lay egg at a time means each cuckoo egg represent a solution, and it put down its egg without any method in other bird nest. The egg with the best quality is carried to the next iteration. The number of nests of other bird is fixed so the cuckoo eggs have the chances of getting forwarded with the probability of (0, 1). The algorithm of cuckoo search is highlighted in Fig. 1 and it's pseudocode is as follows.

- Step 1: Let the population of h host nest be initialize with the value A_n.
- Step 2: Randomly we can get cuckoo egg by Levy flight operation.
- Step 3: Fitness function is calculated with the value T_i.
- Step 4: Select random nest 'j' among h solution.
- Step 5: $T_i \leq T_j$, let i be the current solution
- Step 6: Now replace j as a new solution
- Step 7: Leave a part of the worst nest with P_α, and implementing levy flights operation to built a new nest.
- Step 8: Keep the current best solution
- Step 9: Stop until the criteria is being satisfied
- Step 10: Discover the best nest from the final solution set.

2.2 Levy Flight

Levy flight was introduced by the French mathematician named Paul Levy in 1937. Levy flight is a verbal description of changing of location that stretch further more than the long-established Brownian motion that is being disclosed over hundred years earlier. Initiated from the best acknowledged location, the algorithms propagate a complete modern period at frequent intervals which are arbitrarily distributed accordant to Levy flights [5]. At a specific period of time, the unused generation will be appraised to choose the most prognosticate one. This outgrowth is recurrence until the stopping criteria are being satisfied.

While generating new solutions $x^{(n+1)}$ for cuckoo a Levy flights is accomplished

$$x_a^{(n+1)} = x_a^n + \alpha \oplus \mathrm{Levy}(\lambda) \tag{1}$$

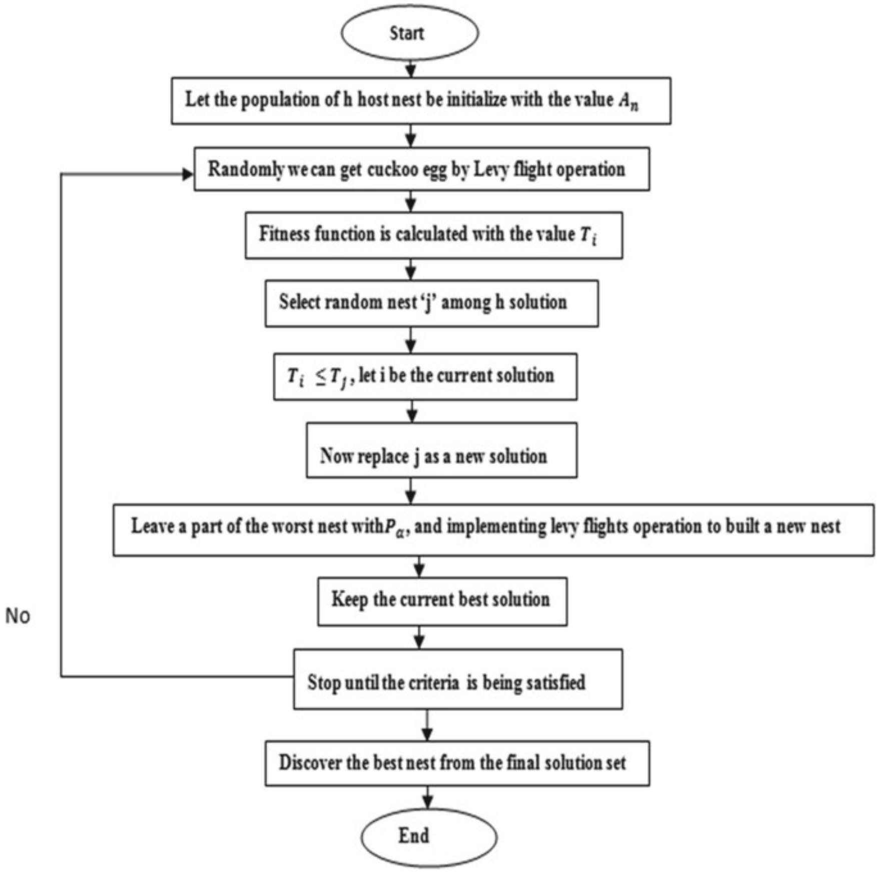


Fig. 1 Flowchart of cuckoo search algorithm

where $\alpha > 0$ is the step size, which indicates obligation that needs to be connected to the layer of the problem. Equation 1 is fundamentally having a random variable equation for random walk. The product $\oplus$ means the element-wise multiplication [6]. The Levy flights basically arrange a random walk while the step length is haggard from levy flights distribution.

$$\text{Levy} \sim u = n^{-\lambda}, \quad (1 < \lambda \leq 3) \tag{2}$$

3 Applications

Cuckoo search is one of the successful swarm optimizations that is being used to solve out different optimization problems [7]. Different applications are being used with cuckoo search such as bloom filter optimization, flood forecasting, hybrid wind gas power, magnetic disk motor, environmental dispatch problem, wireless sensor network, speaker recognition, business optimization, noise cancelation, power network planning, optimized video steganography, bio-level distributed wind generation, image thresholding, speech recognition, optimal power distribution, magnetic flux leakage signals, optimal power flow, etc.

4 Application of Cuckoo Search for Unconstraint Function Optimization

Cuckoo search is one of the successful swarm optimization-based optimization techniques. It has already been proved that cuckoo search is efficient, with its worldwide implementation in various real-world application problems. We have tried to present a simple implementation of cuckoo search algorithm for unconstraint function optimization. For this work, the function to be optimized is considered as the square function.

$$f(x) = \sum_{i=1}^{n} x_i^2 \tag{3}$$

The optimization problem is either a maximization or minimization problem. In this study, $f(x)$ is considered as a minimization problem means the minimum value that is obtained implementing cuckoo search after the final iteration is considered as the optimal value. In the testing phase, CS algorithm is executed with different values of host nest p_a such as host nest $= \{5, 10, 15, 20, 25, 30\}$ and $p_a = \{0.01, 0.02, 0.1, 0.2, 0.4, 0.5\}$. Finally, the optimal value both has been obtained. The parameter setting that has been made for cuckoo search is described in Table 1.

Similar way, some other unconstraint optimization functions are there in which cuckoo search is implemented for optimization. These function along with the optimization environment schedule is highlighted is Table 2. The convergence curve of respective functions ($F1$, $F2$, $F3$) is shown in Fig. 2.

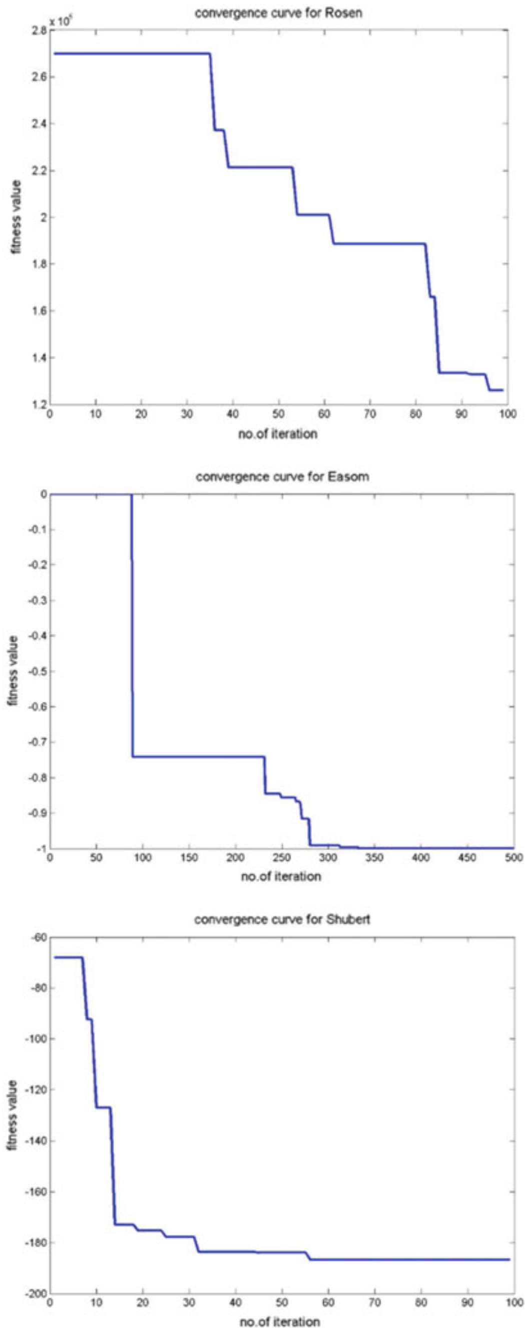
Table 1 Parameter description for CS

Parameter description	Parameter values
Number of host nest	30
Maximum number of iteration	500
p_a	0.25

Table 2 Function along with the optimization environment implementing Cuckoo search algorithm

Sl. no.	Function name	Function description	Range	Actual global minima	Target global minima	Iteration
$F1$	Rosenbrock function	$f(x) = \sum_{i=1}^{d-1} \left[100(x_i + 1 - x_i^2)^2 + (x_i - 1)^2\right]$	(−5,10)	1	1.25	100
$F2$	Easom function	$f(x) = -\cos(x_1)\cos(x_2)\exp\left(-(x_1 - \pi)^2 - (x_2 - \pi)^2\right)$	(−100, 100)	−1	−1	500
$F3$	Shubert function	$f(x) = \left(\sum_{i=1}^{5} i\cos((i+1)x_i + i)\right)\left(\sum_{i=1}^{5} i\cos((i+1)x_2 + i)\right)$	(−10, 10)	−186.7	−186.7	100

Fig. 2 Convergence curve of considered function



References

1. Mohamad, A., Zain, A.M., Bazin, N.E.N., Udin, A.: Cuckoo search algorithm for optimization problems—a literature review. Appl. Mech. Mater. **421**, 502–506 (2013)
2. Pentapalli, V.V.G., Varma, P.R.K.: Cuckoo search optimization and its applications—a review. Int. J. Adv. Res. Comput. Commun. Eng. **5**(11), 2319–5940 (November, 2016)
3. Yang, X.S., He, X.: Bat algorithm: literature review and applications. Int. J. Bio-Inspired Comput. **5**(3)(2013)
4. Davidovi,T., Teodorovi, D., Selmi, M.: Bee colony optimization part I: the algorithm overview. Yugoslav J. Oper. Res. **25**(1), 33–56 (2015)
5. Kamaruzaman, A.F., Zain, A.M., Yusuf, S.M., Udin, A: Levy flight algorithm for optimization problems—a literature review. Appl. Mech. Mater. **421**, 496–501 (2013)
6. Natarajan, A., Subramanian, S.: Bloom filter optimization using cuckoo search. In: International Conference on Computer Communication and Informatics: ICCCI (2012)
7. Xiao, L., Hajjam-El-Hassani, A., Dridi, M.: An application of extended cuckoo search to vehicle routing problem: European Union, pp. 31–35 (2017)

Fruit Fly Algorithm: A Brief Review

Rasmita Rautray, Shreyasi Biswas, Rasmita Dash, and Rajashree Dash

Abstract An optimization algorithm gives an optimum solution by executing iteratively and comparing it with different solutions. Therefore, many optimization algorithms have been discussed and used in all kinds of fields in the past few years. In this study, a brief description of an optimization algorithm called Fruit Fly algorithm (FFOA) is presented. This algorithm is based on the concept of the fruit flies and their extraordinary methods in finding a way to their food. Due to this, it has been applied in many of the research fields like science, engineering, medical, agriculture, etc., in many forms.

Keywords Fruit fly algorithm · Variants · Application area

1 Introduction

The Fruit Fly Optimization Algorithm (FFOA) is a recently proposed swarm intelligence algorithm. This algorithm is based on the concept of the fruit flies and their extraordinary methods in finding a way to their food. Usually, fruit flies live in the mid and the tropical atmospheric zones [1]. They have extremely delicate vision organs which are much better than many different species. They field mainly on spoiled organic products. During the time these flies spend in their nourishment, their special organs smell a wide range of aromas that is noticeable all around. They

R. Rautray (✉) · S. Biswas · R. Dash · R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha, India
e-mail: rashmitaroutray@soa.ac.in

S. Biswas
e-mail: shreyasibiswas1241996@gmail.com

R. Dash
e-mail: rasmitadash@soa.ac.in

R. Dash
e-mail: rajashreedash@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_54

at that point fly towards the areas in which they can sense the food. When they draw near to their nourishment areas, they discover substances utilizing their dreams and after that, they fly towards the course [2]. The original FFOA was modified by some authors so that some of the complex problems like non-bi-level programming problems (NBLPP) can be easily addressed [3]. The improved version of the algorithm is also formed by the combination of two improved fruit fly algorithms. This algorithm is widely used in the fields of science and engineering because of its ease in calculations and also because this algorithm is easily understood. It can also be easily implemented and it can also acquire solutions to many complex problems at a high speed [2]. It has been also applied to solve many complex optimization problems like nonlinear programming problems [3], agricultural output value [4], gasification process [5], etc.

The rest of this paper is sorted out as follows. In Sect. 2, the basic Fruit Fly algorithm is presented. Section 3 details different variants of FFOA. Various application areas of FFOA are highlighted in Sect. 4. Finally, Sect. 5 gives the concluding remarks.

2 Fruit Fly Optimization Algorithm

The first original Fruit Fly Optimization algorithm has been discovered by Pan in the year 2011, which is right now utilized as a free factor multiple times on Science Webs and is chosen as the popularly referred paper in many databases. As of late, the FFOA calculation is utilized in different papers. A special part of the fruit flies is great at gathering smells gliding noticeable all around and can even stiff out the substances that are sourced around 40 km away. At that point, the fruit flies can likewise identify the area where the nourishment and different flies are by virtue and different flies by virtue of a pointed vision in the wake of flying close to the particular target. Afterward, the fly could likewise discover that area where the nourishment is present [6]. The detailed steps and workflow of Fruit Fly algorithm are discussed in the next subsection.

2.1 Steps of Fruit Fly Algorithm

The following are the steps that are included in the Fruit Fly optimization algorithm. These steps are taken into use when this algorithm is used while performing in various application areas. The steps are as follows:

- The fruit fly swarm's underlying position is haphazardly created.

$$\text{Ini}X\_\text{axis} \tag{1}$$

$$\text{Ini}Y\_\text{axis} \tag{2}$$

- By utilizing its feeling of smell, the fruit fly haphazardly begins scanning for its nourishment.

$$X_i = X\_\text{axis} + \text{arbitrary esteem} \tag{3}$$

$$Y_i = Y\_\text{axis} + \text{arbitrary esteem} \tag{4}$$

As the situation of the sustenance isn't known, the separation (Dist) to the cause is first evaluated, and the estimation of smell focus (judged), started as (S), which is the opposite of the separation, is then ascertained.

$$\text{Dist}_i = \sqrt{X_i^2 + Y_i^2} \tag{5}$$

$$S_i = 1/\text{Dist}_i \tag{6}$$

- The passed judgment on estimations of little fixation (S) is substituted into the smell focus judge work, then we get the smell fixations

$$\text{Smell}_i = \text{Func}(S_i) \tag{7}$$

- The Fruit Fly whose position has the best smell fixation is recognized.

  (For example—Using the base esteem)

$$[\text{Best} - \text{the} - \text{Smell} - \text{Best} - \text{the} - \text{Index}] = \text{minimum (Smell)} \tag{8}$$

- A best smell fixation esteem as well as x, y organize is kept;

An organic product swarm would notice the position of the nourishment and move to the position.

$$\text{Smells} - \text{the} - \text{best} = \text{best} - \text{the} - \text{Smell} \tag{9}$$

$$X - \text{axis} = X(\text{Best} - \text{Index}) \tag{10}$$

$$Y - \text{axis} = X(\text{Best} - \text{Index}) \tag{11}$$

Too much advancement is entered and the means 2–5 are rehashed and afterward, it is made a decision about whether the smell focus is higher than that in the past emphasis [4].

3 Variants of Fruit Fly Algorithm

Table 1 illustrates the variants of Fruit Fly Algorithm Variants may include the form which contains the improved version to tackle the limitations of the original FFOA.

4 Application Areas of FFOA

FFOA is a very robust algorithm and because of its way of calculating and solving different types of complex optimization problems, it has been accepted all over. FFOA is applied in various areas like science, engineering, medical, agriculture, and many more. As we all know that it works on the concept of the fruit flies that can sense their nourishment from a distance and move towards it, in the same way, FFOA works. So, below are some of the areas or fields where the FFOA has been applied (Table 2).

5 Conclusion

This paper highlights the fruit fly algorithm and its use in many of the areas. FFOA is the algorithm that is inspired by the food searching behavior of the natural flies and how they can sense their food from far away areas and then chase for their nourishment. In many of the areas, the traditional form of the algorithm has also been revised so that it can be used in solving various complex problems related to various optimization problems. In many fields like, mathematical, engineering, and many biological and scientific areas, there are many new fields that have been discovered. So keeping in mind this fact, FFOA is also being improved day by day so that it can cope up with the various optimized problems arising very frequently. This paper is detailed about the FFOA and the steps that are to be performed sequentially to solve any issues. The paper also contains the various variations and the traditional forms of FFOA that has been included in various papers and also the application areas where this algorithm is used.

Table 1 Variants of FFOA

Sl. No.	Variants of FFOA used in different applications	References
1.	In this paper, a novel calculation is developed for illuminating a nonlinear bi-level programming issue dependent on another smart calculation, to be specific the organic product fly advancement calculation	[3]
2.	A new improved FFOA named IAFFOA is described to tackle the high-dimensional function optimization problems and engineering optimization problems	[7]
3.	FFOA is applied in the UAV (unnamed aerial vehicles) path planning which simultaneously presents the improved version of the FFOA	[8]
4.	In this examination, we propose an enhanced organic product fly advancement calculation (FOA) in view of straight reducing step and strategic mayhem mapping (named DSLC-FOA) for settling benchmark work unconstrained enhancement issues and obliged auxiliary building plan enhancement issues	[9]
5.	Another Fruit Fly optimization algorithm in light of an ongoing report on how well organic product fly's small cerebrum discovers nourishment was created. The standard Fruit Fly optimization algorithm was improved by presenting the postponement and visual element discovery stages that describe an organic product fly's sustenance seek technique	[10]
6.	In this examine, an enhanced organic product fly improvement calculation is exhibited to illuminate the MKP (multi-dimensional knapsack) problem	[11]
7.	This examination shows usage of later created calculation to be specific FOA on 2DSPP (Two–dimensional strip packing problem)	[12]
8.	In this investigation, another altered FOA is proposed for settling the outstanding voyaging businessperson issue (TSP) which is a standout amongst the most concentrated discrete improvement issues	[13]
9.	In this paper, by breaking down FOA, we demonstrated that the combination of FFOA relies upon the underlying area of the swarm and that the irregularity of the underlying area of the swarm more often than not results in a nearby ideal. To address this issue, we proposed MSFOA, a novel multi-scale helpful transformation Fruit Fly Optimization calculation	[14]
10.	The principle commitment of this paper is to display a MILP display and propose viable two-arrange learning-based organic product fly enhancement calculation for the UPMSP_RC	[15]
11.	In this investigation, an as good as ever FOA called LP–FOA is structured to settle high-dimensional and complex nonstop advancement issues and combinatorial streamlining issues	[16]
12.	In this paper, we have acquainted a novel FFO calculation with take care of the range picture enrolment issue	[17]
13.	In this paper, a novel calculation named fruit fly optimization clustering algorithm (FOCA), is proposed to recognize dynamic protein buildings by consolidating FFOA and quality articulation profiles	[18]
14.	To move forward the assembly execution of the FFOA, an enhanced FFOA is proposed in this paper by presenting another arrangement creating technique dependent on ordinary cloud generator and a parameter versatile methodology	[19]

(continued)

Table 1 (continued)

Sl. No.	Variants of FFOA used in different applications	References
15.	This paper exhibits a compelling and enhanced FFOA (IFFOA) for improving numerical capacities and unravelling joint recharging issues	[20]

Table 2 Various application areas of FFOA

Sl. No.	Application areas of FFOA	References
1.	Slant seek technique and worldwide searching	[1]
2.	Nonlinear bi-level programming problem (NBLLP)	[3]
3.	Agricultural	[4]
4.	Engineering and mathematical	[7]
5.	Autonomous path planning	[8]
6.	Structural engineering design	[9]
7.	Pot hole harm moderation	[10]
8.	Knapsack problem	[11]
9.	Engineering and mathematical	[12]
10.	Discrete optimization	[13]
11.	Swarm optimization issue	[14]
12.	Resource utilization	[15]
13.	High-dimensional optimization	[16]
14.	Range image registration	[17]
15.	Multiple-skilled resource	[21]

References

1. Han, X., Liu, Q., Wang, H., Wang, L.: Novel fruit fly optimization algorithm with trend search and co-evolution. Knowl. Based Syst. **141**, 1–17 (2018)
2. Pan, Q., Sang, H., Duan, J., Gao, L.: An improved fruit fly optimization algorithm for continuous function optimization algorithm. Knowl. Based Syst. **62**, 69–83 (2014)
3. Wang, G., Ma, L., Chen, J.: A bi-level improved fruit fly optimization algorithm for the nonlinear bi-level programming problem. Knowl. Based Syst. **138**, 113–223 (2017)
4. Han, S., Pan, W., Zhou, Y., Liu, Z.: Construct the prediction model for China agricultural output value based on the optimization neural network of fruit fly optimization algorithm. Futur. Gener. Comput. Syst. **86**, 663–669 (2018)
5. Niu, J., Zhong W., Liang y., Luo, N., Qian, F.: Fruit fly optimization based on differential evolution and its application on gasification process operation optimization. Knowl. Based Syst. **88**, 253–263 (2015)
6. Shen, L., Chen, H., Yu, Z., Kang, W., Zhang, B., Li, H., Yang, B., Liu, D.: Evolving support vector machines using fruit fly optimization for medical data classification. Knowl. Based Syst. **96**, 61–75 (2016)
7. Wu, L., Liu, Q., Tian, X., Zhang, J., Xiao, W.: A new improved fruit fly optimization algorithm IAFOA and its application to solve engineering optimization problems. Knowl. Based Syst. **144**, 153–173 (2018)

8. Zhang, X., Lu, X., Jia, S., Li, X.: A novel phase angle-encoded fruit fly optimization algorithm with mutation adaptation mechanism applied to UAV path planning. Appl. Soft Comput. **70**, 371–388 (2018)
9. Du, T., Ke, X., Liao, J., Shen, Y.: DSLC-FOA: Improved fruit fly optimization algorithm for application to structural engineering design optimization problems. Appl. Math. Model. **55**, 314–339 (2018)
10. Kanarachos, S., Diyzqah, A., Chrysakis, G.: Optimal design of a quadratic parameter varying vehicle suspension system using contrast-based fruit fly optimisation. Appl. Soft Comput. **62**, 463–477 (2018)
11. Meng, T., Pan, Q.: An improved fruit fly optimization algorithm for solving the multidimensional knapsack problem. Appl. Soft Comput. **50**, 79–93 (2017)
12. Babaoglu, I.: Solving 2D strip packing problem using fruit fly optimization algorithm. In: 8th International Conference on Advances in Information Technology, IAIT2016, 19–22 December 2016, Macau, China, vol. 111, pp. 52–57 (2016)
13. Iscan, H., Gunduz, M.: An application of fruit fly optimization algorithm for travelling salesman problem. In: 8th International Conference on Advances in Information Technology, IAIT2016, 19–22 December, Macau, China, vol. 111, pp. 58–63 (2016)
14. Zhang, Y., Cui, G., Wu, J., Pan, W., He, Q.: A novel multi-scale cooperative mutation fruit fly optimization algorithm. Knowl. Based Syst. **114**, 24–35 (2016)
15. Zheng, X., Wang, L.: A two-stage adaptive fruit fly algorithm for unrelated parallel machine scheduling problem with additional resource constraints. Expert Syst. Appl. **65**, 28–35 (2016)
16. Wang, L., Liu, R., Liu, S.: An effective and efficient fruit fly optimization algorithm with level probability policy and its applications. Knowl. Based Syst. **97**, 158–174 (2016)
17. Li, T., Gao, L., Li, P., Pan, Q.: An ensemble fruit fly optimization algorithm for solving range image registration to improve quality inspection of free-form surface parts. Inf. Sci. **367–368**, 953–974 (2016)
18. Lei, X., Ding, Y., Fujita, H., Zhang, A.: Identification of dynamic protein complexes based on fruit fly optimization algorithm. Knowl. Based Syst. **105**, 270–277 (2016)
19. Wu, L., Zuo, C., Zhang, H.: A cloud model based fruit fly optimization algorithm. Knowl. Based Syst. **89**, 603–617 (2015)
20. Wang, L., Shi, Y., Liu, S.: An improved fruit fly optimization algorithm and its application to joint replishment problems. Expert Syst. Appl. **42**, 4310–4323 (2015)
21. Wang, L., Zheng, X.: A knowledge-guided multi-objective fruit fly optimization algorithm for the multi-skill resource constrained project scheduling problem. Swarm Evol. Comput. **38**, 54–63 (2018)

Medical Data Classification Using Jaya Optimized ELM

Prajna Paramita Debata and Puspanjali Mohapatra

Abstract The implication of machine learning approaches in healthcare issues makes the diagnosis and prognosis of cancer like deadly diseases easier and affordable. The sole aim of machine learning researchers is to design a robust classification model that can classify the medical data with high accuracy and less computational complexity. Here, Jaya optimized extreme learning machine (Jaya-ELM) is proposed for classifying medical data. To estimate the efficacy of the proposed model, two benchmark medical datasets, Lung Cancer and Breast Cancer, are collected from the UCI repository. The classification accuracy percentage is taken as a performance evaluation measure in this model. The result of this model is compared with the existing Neural Network (NN), Jaya optimized NN (Jaya-NN) and ELM models. The experimental result shows that the Jaya-ELM is significantly better than other models.

Keywords Machine learning techniques · Medical data · Classification model · Jaya optimization algorithm · Extreme learning machine

1 Introduction

In this era, the most demanding task of data mining is the classification of highly complex data which consists of a large number of samples and attributes. In the field of pattern classification like text and image classification [1], optical character recognition, face recognition [2], protein sequence classification [3], medical data classification [4] and biomedical image classification [5], classification performs a

P. P. Debata (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhuabneswar, Odisha, India
e-mail: prajnaparamitadebata@soa.ac.in

P. Mohapatra
Department of Computer Science and Engineering, International Institute of Information Technology, Bhubaneswar, Odisha, India
e-mail: puspanjali@iiit-bh.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_55

major role. This paper focuses on classification of medical data. Many researchers have developed different statistical models for medical data classification, still, it is challenging due to some issues [6] such as (i) selection of relevant features is a non-deterministic polynomial-time (NP)-Hard problem (ii) gene expression datasets deal with curse of sparsity (i.e. publicly available microarray datasets contain very few number of samples) and (iii) genes in gene expression datasets are highly correlated and interrelated with each other. According to GLOBOCAN database [7] and the World Health Organization [8], cancer like deadly diseases is threatening human species due to the rapid change of environment. Some of these diseases can be effectively cured for lifetime by early diagnosis, if not cured then it can be treatable. So, to help the medical practitioners, machine learning researchers try to design intelligent and robust classification models for diagnosing these diseases.

Numerous statistical and soft computing based classification approaches have been applied in the classification field. The major drawback of statistical methods [9, 10] is that these methods are not efficient in case of nonlinear problems. Like statistical methods, soft computing methods also have some merits and demerits. The drawbacks of soft computing approaches [11] are: training time of high dimensional datasets is very high, selecting an optimum kernel function is quite difficult and the final model depends on variable weights which makes it difficult to interpret. According to the literature survey, SVM [9, 10] and Artificial Neural Network (ANN) have been applied by the researchers usually and extensively. In SVM, it is difficult to fine-tune the hyper-parameters like gamma and Cost-*C*. It is also not easy to view the impact of these parameters. In ANN, backpropagation algorithm is generally used as a learning technique and the major demerit of it is very slow rate of convergence. So, various optimization techniques are applied to optimize ANN.

In order to achieve high performance, traditional feed forward network tunes the weights and biases at different layers in each iteration. Hence, large number of iteration steps are required. If the learning method uses the gradient descent learning algorithm then it is easily trapped at local minima due to randomly chosen parameters. Generally, Single-hidden Layer Feed Forward Neural Network (SLFN) needs maximum no. of hidden neurons, randomly chosen input biases and weights, to be tuned each time. To avoid this problem, Huang et al. [12] introduced a revolutionary model, called Extreme Learning Machine (ELM) which has good generalization ability and faster learning speed. The randomly chosen parameters of ELM such as weights and biases can make its outcome suboptimal. For optimizing the random parameters of ELM, many meta-heuristic learning algorithms are applied [13, 14] and succeeded in achieving better accuracy. In this paper, a simple but powerful Jaya optimization algorithm is integrated with ELM to classify medical data. This algorithm is proposed by Venkata Rao [15] to solve both constrained and unconstrained optimization problems based on the conception of achieving the best solution and discarding the worst solution. Here a comparison is made between the classification accuracy of NN, Jaya optimized NN (Jaya-ELM), ELM and Jaya optimized ELM. To evaluate this model two benchmark cancer datasets such as Breast Cancer and Lung cancer are considered. In Sect. 2, the model description part is covered and in

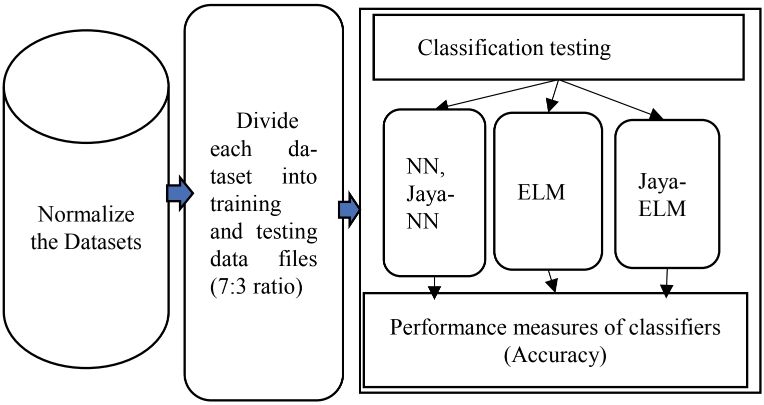


Fig. 1 Abstract view of medical data classification

Sect. 3, all the supported methodologies are discussed. Section 4 and 5 reflect the experimentation and result in the analysis part, respectively.

2 Abstract View of the Model

Each data set is normalized using Eq. (1) between the range (−1, 1).

$$Y_n = m + (n - m) * \left(y - X_{\text{min}}\right) / \left(X_{\text{max}} - X_{\text{min}}\right) \tag{1}$$

Here, y= original value, Y_n = normalized form of y, $m = -1$ and $n = 1$, X_{min} = minimum value of the dataset, X_{max} = maximum value of the dataset.

After normalizing the datasets, divide each dataset into training and testing data files in the ratio of 7:3. Then compare the classification accuracy of NN, Jaya-NN, ELM and Jaya-ELM with two datasets. The abstract view of the model is given in Fig. 1.

3 Methods Used

3.1 Neural Network (NN) and Its Advantages

Here, the applied NN model is divided into three layers such as input layer, hidden layers and output layer. Input layer accepts data, features and signals from the exterior environment. Hidden layers comprise of neurons which helps in extracting patterns related with the system to be analyzed. In this layer, all the internal processing occurs.

The neurons of these three layers are completely interconnected with each other. The final outputs are obtained in the output layer from the processing conducted by the neurons of the hidden layers. This network model is effective in the area of classification and pattern recognition [16]. Neural Network model is convenient than other conventional models [16] because the decision-making process of this model relies on the aggregation of entire input patterns.

3.2 Jaya Optimization Algorithm

The major advantage of Jaya optimization [15] algorithm is that it requires less algorithm-specific parameters. The steps of Jaya algorithm are summarized as follows:

Step 1: Initialize the size of the population, number of design variables and the termination criteria.
Step 2: From the population, the best and worst solutions should be found out.
Step 3: The result based on the best and worst solutions would be changed by using the Eq. (2).

$$O'_{j,k,i} = O_{j,k,i} + r_{1,j,i}\left[\left(O_{j,\mathrm{best},i}\right) + \left|\left(O_{j,k,i}\right)\right|\right] - r_{2,\mathrm{best},i}\left[\left(O_{j,\mathrm{worst},i}\right) - \left|\left(O_{j,\mathrm{worst},i}\right)\right|\right] \quad (2)$$

During ith iteration, $O_{j,\,k,\,i}$ is the value of the ith variable for the kth candidate. Here, k is the population size, i value gives the count of iteration and j value gives the count of design variables.
Step 4: Then the existing solution is compared with the modified one and if it is found that the modified solution is better, then it will be replaced by the previous one otherwise the previous solution will be kept.
Step 5: The procedure, from step 2 to step 4 will be repeated till the termination criteria is reached.

3.3 Elm

The basic structure of ELM is shown in Fig. 2. The mechanism of ELM is summarized as follows:
Step 1 Randomly select the input weights (w_i) and hidden layer biases (b_i).
Step 2 Calculate output matrix H of the hidden layer.
Step 3 Apply the equation $\hat{\beta} = H^{\dagger}T$ to compute the output weight (β)

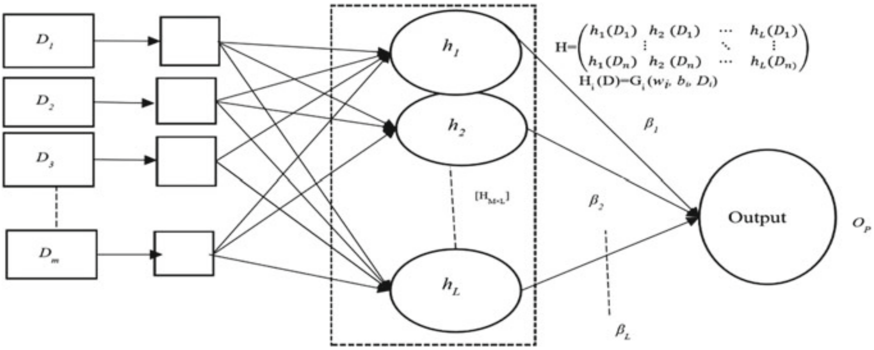


Fig. 2 Basic structure of ELM

3.4 The Proposed Jaya-ELM Algorithm

Algorithm:	Medical data classification by using Jaya-ELM model
Input:	Dataset; population size (*PS*); Hidden layer size (H_c)
Output:	Classification accuracy percentage
Description	$newP_i$ is the new population, Obj_i is the objective value of the new population, *W* is the weight vector, *train_acc_percentage* and *test_acc_percentage* are accuracy percentage of *train_data* and *test_data,* respectively
Step 1	Divide the dataset into *train_data* (*training_input, training_output*) and *test_data* (*testing_input, testing_output*) in the ratio of 7:3
Step 2	Generate H_c no. of random weight population, each having size of $1 \times H_c$ $P_i = \{W_1^i, W_2^i, W_3^i, \ldots, W_{H_c}^i\}$ for $i = 1, 2, 3, \ldots$PS
Step 3	For each population $P_{i,}$ find the *error value* or *miss_classification_rate* in ELM by Step 4 to Step 8
Step 4	For each *training_input* find
Step 5	$H = training\_input \times P_i$
Step 6	$\beta_i = pseudo\_inverse\ (H) \times training\_output$
Step 7	$obtained\_output = (testing\_input \times P_i) \times \beta_i$
Step 8	Obj_i is calculated from *miss_classification_rate* which is estimated by comparing *obtained_output* and *testing_output*
Step 9	Find $g\_best\ P_i$ where $arg_{min}\ (Obj_i)$
Step 10	Find $g\_worst\ P_i$ where $arg_{max}\ (Obj_i)$
Step 11	For each population P_i $new\ P_i = P_i + rand\ (1, H_c) \times (g\_best - \lvert P_i \rvert) - rand\ (1, H_c) \times (g\_worst - \lvert P_i \rvert)$
Step 12	Find $newObj_i$ by Step 4 to Step 8
Step 13	If $Obj_i > newObj_i$
Step 14	Replace P_i with $new\ P_i$

(continued)

(continued)

Algorithm:	Medical data classification by using Jaya-ELM model
Step 15	Repeat Step 9 to Step 14, till the termination criteria reaches
Step 16	*g_best* is considered as the final weight W_{g_best} and keep the corresponding β_{g_best}
Step 17	$Calculated\_output = (testing\_input \times W_{g\_best}) \times \beta_{g\_best}$ *and* determine the *actual label* of the class for both train_data and test_data
Step 18	*miss_classification_rate* is obtained by comparing *actual label* and *expected label* of the class for both train_data and test_data
Step 19	$train\_acc\_percentage = (1 - miss\_classification\_rate)/sizeof(train\_data)$
Step 20	$test\_acc\_percentage = (1 - miss\_classification\_rate)/sizeof(test\_data)$

4 Experimentation and Dataset Description

In this paper, two benchmark binary class cancer datasets such as Breast Cancer (699 samples and 9 features) and Lung cancer (32 samples and 56 features) are collected from UCI repository for evaluating the proposed model. Training accuracy and testing accuracy are calculated from ELM and Jaya-ELM for each dataset in 10–100 number hidden nodes (NH) with an increment of 10 neurons each time. **Jaya parameters setting:** In this study, the value of population size and the number of iterations are taken as 100. **NN parameters setting:** In this study, the number of hidden layers is taken as 3 and, in each layer, five nodes are considered.

4.1 Experiment 1 with Breast Cancer Dataset

Table 1 enlists the training and testing accuracy of ELM and Jaya-ELM. Based on the result of Table 1, Figs. 3 and 4 are plotted which display the comparison between the training and testing accuracy of ELM and Jaya-ELM.

4.2 Experiment 2 with Lung Cancer Dataset

Table 2 enlists the training and testing accuracy of ELM and Jaya-ELM. Based on the result of Table 2, Figs. 5 and 6 are plotted which display the comparison between the training and testing accuracy of ELM and Jaya-ELM.

Table 1 Comparison of ELM, Jaya-ELM training accuracy (Tra) and testing accuracy (Tsa) in breast cancer dataset

NH	ELM		Jaya-ELM	
	Tra	Tsa	Tra	Tsa
10	0.9635	0.9492	0.9643	0.9572
20	0.9628	0.9425	0.9705	0.9505
30	0.9737	0.9511	0.9703	0.9591
40	0.9742	0.9542	0.9754	0.9622
50	0.9717	0.9591	0.9763	0.9671
60	0.9752	0.9582	0.9791	0.9662
70	0.9768	0.9531	0.9783	0.9611
80	0.9762	0.9612	0.9805	0.9592
90	0.9762	0.9636	0.9811	0.9646
100	0.9792	0.9658	0.9825	0.9691

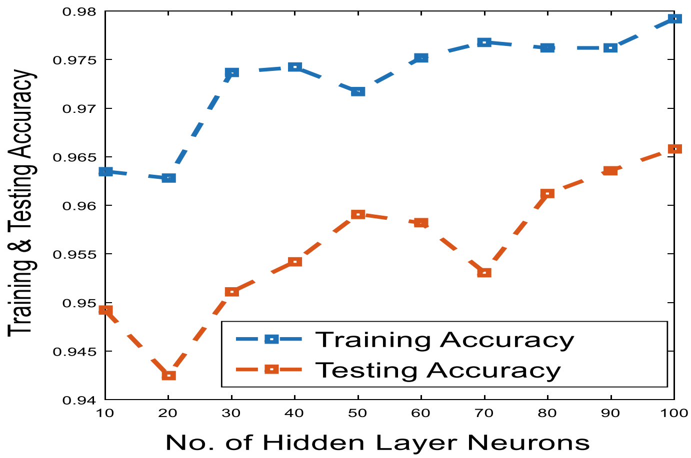


Fig. 3 NH versus training and testing accuracy in ELM (Breast Cancer)

5 Result Analysis

Here, a popularly used meta-heuristic Jaya algorithm optimized ELM (Jaya-ELM) model is evaluated by two medical datasets: Breast Cancer and Lung Cancer. The results, obtained by the evaluation, are compared with other models like NN, Jaya-NN and ELM. Training accuracy and testing accuracy are taken as the performance evaluation measures, to compare and test the robustness and stability of the proposed Jaya-ELM model. It is clearly visualized from Tables 1 and 2 that Jaya-ELM needs a smaller number of hidden neurons than that of ELM to achieve similar training accuracy. In Table 3, a comparison between maximum training and testing accuracies

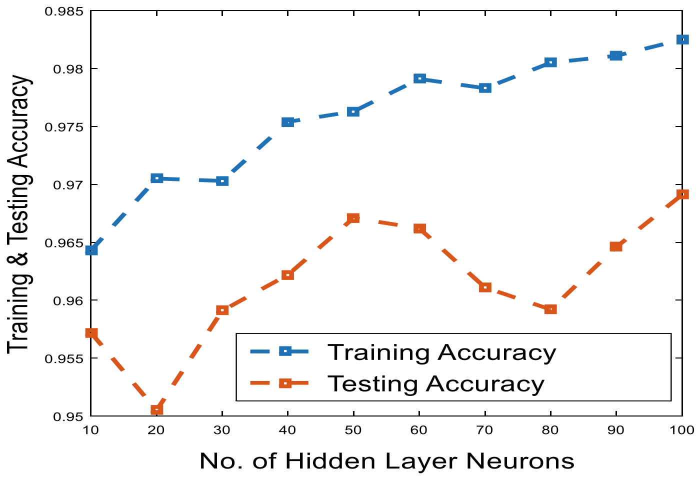


Fig. 4 NH versus training and testing accuracy in Jaya-ELM (breast cancer)

Table 2 Comparison of ELM, JAYA-ELM training accuracy (Tra) and testing accuracy (Tsa) in lung cancer dataset

NH	ELM		Jaya-ELM	
	Tra	Tsa	Tra	Tsa
10	0.9293	0.9043	0.9303	0.9093
20	0.9237	0.9026	0.9244	0.9076
30	0.9069	0.8948	0.9093	0.8998
40	0.9135	0.9045	0.9046	0.8995
50	0.9283	0.8998	0.9353	0.9008
60	0.9331	0.9083	0.9384	0.8983
70	0.9372	0.9107	0.9407	0.8917
80	0.9387	0.9056	0.9464	0.8906
90	0.9392	0.9134	0.9516	0.9094
100	0.9401	0.9173	0.9533	0.9123

of NN, Jaya-NN, ELM and Jaya-ELM is shown. It suggests the Jaya-ELM gives a significantly better result than NN, Jaya-NN and ELM.

6 Conclusion and Future Work

Here, Jaya-ELM is tested for the classification of two small medical datasets (with less sample size and features). More than that, other classification models like ELM,

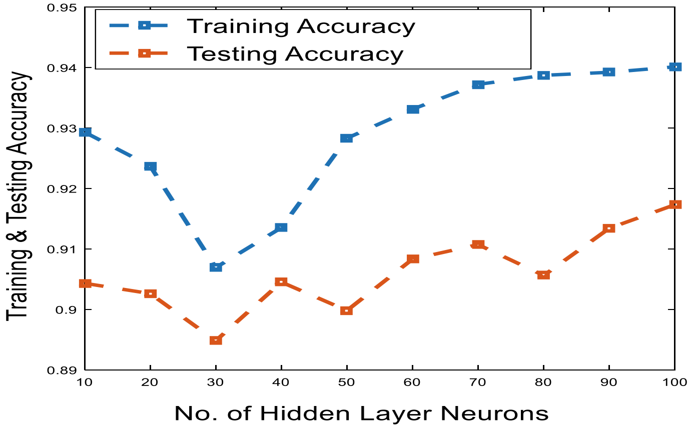


Fig. 5 NH versus training and testing accuracy in ELM (Lung cancer)

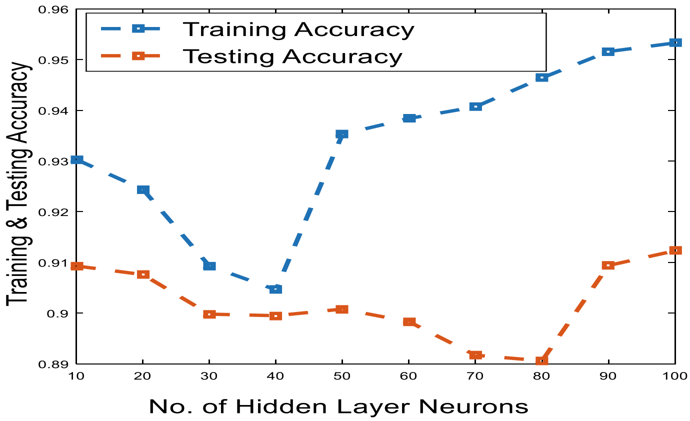


Fig. 6 NH vs. Training & Testing Accuracy in Jaya-ELM (lung cancer)

Table 3 Comparison between maximum training and testing accuracies of NN, Jaya-NN, ELM and Jaya-ELM in Breast Cancer (BC) and lung cancer (LC) datasets

Dataset	NN		Jaya-NN		ELM		Jaya-ELM	
	Tra	Tsa	Tra	Tsa	Tra	Tsa	Tra	Tsa
BC	0.8165	0.7278	0.9178	0.7637	1	0.9801	1	0.9881
LC	0.7983	0.7452	0.8849	0.8169	0.9848	0.9348	1	0.9398

Jaya-NN and NN are also discussed and compared with this model. In this paper, ELM is combined with Jaya to make the classification of medical data, more stable and robust. The findings reveal that Jaya-ELM outperforms the other models. This work suggests that the proposed Jaya-ELM model can be applied successfully for classifying high dimensional gene expression data.

References

1. Merkl, D.: Text classification with self-organizing maps: Some lessons learned. Neurocomputing **21**(1–3), 61–77 (1998)
2. Abbas, E. I., Safi, M. E., Rijab, K. S.: Face recognition rate using different classifier methods based on PCA. In: 2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT), pp. 37–40, IEEE (2017)
3. Chen, L., Zhang, Y.H., Huang, G., Pan, X., Wang, S., Huang, T., Cai, Y.D.: Discriminating cirRNAs from other lncRNAs using a hierarchical extreme learning machine (H-ELM) algorithm with feature selection. Mol. Genet. Genomics **293**(1), 137–149 (2018)
4. Ahmad, F., Isa, N.A.M., Hussain, Z., Osman, M.K., Sulaiman, S.N.: A GA-based feature selection and parameter optimization of an ANN in diagnosing breast cancer. Pattern Anal. Appl. **18**(4), 861–870 (2015)
5. Vardhana, M., Arunkumar, N., Lasrado, S., Abdulhay, E., Ramirez-Gonzalez, G.: Convolutional neural network for bio-medical image segmentation with hardware acceleration. Cogn. Syst. Res. **50**, 10–14 (2018)
6. Bolón-Canedo, V., Sánchez-Marono, N., Alonso-Betanzos, A., Benítez, J.M., Herrera, F.: A review of microarray datasets and applied feature selection methods. Inf. Sci. **282**, 111–135 (2014)
7. Parkin, D.M., Bray, F., Ferlay, J., Pisani, P.: Estimating the world cancer burden: Globocan 2000. Int. J. Cancer **94**(2), 153–156 (2001)
8. World Health Organization.: Research for International Tobacco Control. WHO report on the global tobacco epidemic, 2008: The MPOWER package. World Health Organization (2008)
9. Varatharajan, R., Manogaran, G., Priyan, M.K.: A big data classification approach using LDA with an enhanced SVM method for ECG signals in cloud computing. Multimed. Tools Appl. **77**(8), 10195–10215 (2018)
10. Ye, F.: Evolving the SVM model based on a hybrid method using swarm optimization techniques in combination with a genetic algorithm for medical diagnosis. Multimed. Tools Appl. **77**(3), 3889–3918 (2018)
11. Duda, P.O., Hart, P.E.: Pattern Classification and Scene Analysis. Willey, New York (1973)
12. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: A new learning scheme of feedforward neural networks. Neural Netw. **2**, 985–990 (2004)
13. Mohapatra, P., Chakravarty, S., Dash, P.K.: An improved cuckoo search based extreme learning machine for medical data classification. Swarm Evol. Comput. **24**, 25–49 (2015)
14. Mohapatra, P., Chakravarty, S., Dash, P.K.: Microarray medical data classification using kernel ridge regression and modified cat swarm optimization based gene selection system. Swarm Evol. Comput. **28**, 144–160 (2016)
15. Rao, R.: Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Ind. Eng. Comput. **7**(1), 19–34 (2016)
16. Mohammed, M.A., Ghani, M.K.A., Arunkumar, N.A., Hamed, R.I., Abdullah, M.K., Burhanuddin, M.A.: A real time computer aided object detection of nasopharyngeal carcinoma using genetic algorithm and artificial neural network based on Haar feature fear. Futur. Gener. Comput. Syst. **89**, 539–547 (2018)

Application of Backward Elimination in Multiple Linear Regression Model for Prediction of Stock Index

Chandan Mishra, Lagnajeet Mohanty, Smita Rath, Rahul Patnaik, and Rajesh Pradhan

Abstract Technological progress raises the stock and shares market analysis. Specific mathematical and machine learning methods improve decision-making. Major research work focused on the stock price forecast feature based on historical rates and volume. In this work, TCS stock index analyzes the performance measurements using statistical methods in Python environment. In this analysis, the results obtained are superior to the existing methods. The methods for analyzing the financial market are based on a multiple linear regression using backward elimination method. This paper focuses on the best independent variables to forecast the stock market's closing price. This research is used to find specific variables that show the greatest effect on closing price prediction.

Keywords Stock market · Linear regression · Multiple linear regression · Backward elimination technique

1 Introduction

A stock (also known as "shares" or "equity") is a type of security that signifies partial ownership in the issuing company. This entitles the stockholder to that proportion of

C. Mishra · L. Mohanty · S. Rath (✉) · R. Patnaik · R. Pradhan
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to Be University), Bhubaneswar, Odisha, India
e-mail: smitarath@soa.ac.in

C. Mishra
e-mail: mishrachandan1997.cm@gmail.com

L. Mohanty
e-mail: lagnajeet2000@gmail.com

R. Patnaik
e-mail: patnaik.rahul2503@gmail.com

R. Pradhan
e-mail: raj4568111@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_56

the company's assets and earnings. It is a place where shares are traded in publicly traded companies. The primary market is where businesses in an initial public offering (IPO) give out their shares to the general population to secure funding to increase their assets. The idea behind the operation of the stock market is quite easy. It functions as an auction house, and the stock market provides stock buyers and sellers with a forum for negotiating prices and making profitable trades.

In history, stock trading has probably occurred in a physical marketplace pretty much like the sale of food and furniture. These days, through Internet and online stock brokers, the stock market is working digitally. Each exchange occurs on a market-by-stock basis, but due to media, political issues, economic indicators, and other factors, net share prices frequently change unexpectedly. Stock price prediction is a crucial topic in economics and finance that has gained the attention of many peoples and data scientists over the years to design better forecasting models. Forslund and Akesson [1] proposed a multiple linear regression model to predict closing share price for 44 companies and the regression was done in Microsoft Excel using a built-in LINEST function. Seethalakshmi [2] used linear regression for financial market analysis combined with a statistical method in the R environment to develop a stock market predictor model that can perform better than the existing model. Kumari and Yadav [3] did a linear regression analysis study and found a way to perform its calculations in SPSS and Excel. Dong et al. [4] created an author age prediction technique from text using linear regression model based on shallow text features. They used both new features as POS patterns as well as content-oriented ones and took data from blogs, telephone conversations, and online forum posts. Mekparyup et al. [5] innovated a multiple regression model for estimating the amount of nitrogen oxides in an area on a monthly basis taking wind-speed and direction air temperature, and relative humidity and dependent variable as nitrogen oxide. Faridah Hani Mohamed et al. [6] devised a multiple regression model for recreation of gene regulatory networks to solve cascade error problems which gave a satisfactory result with AUROC values above 0.5. Supatra and Putro [7] designed a model based on linear regression to estimate thunderstorm activity which takes six inputs and gives one output with below 50% error and maximum epoch reaching up to 1000 iterations. Altay and Satman [8] induced artificial neural network and linear regression in their stock market forecasting and compared the results and found out that ANN models performed much better than regression models. Ichise et al. [9] made up strategies to improve neuroreceptor parameter estimation by linear regression analysis. Vincenzo et al. [10] developed an electricity consumption forecasting using a linear regression model to get a long term estimation. They also forecasted GDP per capita and price elasticity's of domestic and non-domestic for daily consumption of electricity.

A stock market comprises of the following elements. Close price per share is the last market price by any stock buyer for a share of such a stock that lasts the operating hours of the stock trading exchange. Opening price—the opening price, like the closing price, is the value from a market day very next transaction. The high price in a given time period is the highest price of the stock index. Low price is the minimum price in a given time period. Some traditional techniques are already being used for predicting purposes. There are numerous models which are used by people but here are some of the honorable mentions below.

1.1 ARIMA Model

ARIMA stands for the unified moving average of autoregressive technique. It is a template class that captures in time series information a suite of different standard temporal structures. ARIMA uses multiple lagged time series measurements to estimate observations. Its main role is short-term forecasting which requires at least minimum historical data points to calculate. But using only **ARIMA model**, it is very difficult to **model** the nonlinear relationships between variable.

1.2 Theory of Random Walk Model

The random walk model is one of the simplest but most important models in time series forecasting. This model suggests that the variable takes a random step away from its preceding value in each time span and that the steps are distributed in size independently and identically.

1.3 Regression

Regression techniques are one of the most popular statistical techniques used for predictive modeling and data mining tasks. Used primarily to model and analyze the relationships between variables and often how they make a contribution and are associated with producing a specific result together. Its applications are in the fields of banking, investment, and other fields of study that try to assess the strength of the connection between one dependent variable (usually denoted by Y) and a series of other (non-dependent) changing variables.

There are various types of regression but the simplest one of them is linear regression.

1.4 Linear Regression

A linear regression refers to a method that represents linear variables. In statistics, this method is a linear technique to model the connection among a scalar variable and one or more explanatory variables. There are two types of linear regression simple linear and multiple linear regressions.

1.4.1 Simple Linear Regression

Simple linear regression is skillful for finding a relationship between two continuous variables. It takes as input two-dimensional sample points with one independent variable and one dependent variable and considers a linear component predicting the dependent values as a parameter of the predictor variables.

It works on the formula given by the relation

$$Y = \alpha + \beta X + \varepsilon \tag{1}$$

where β is the estimate of the slope coefficient, ε is the random error and X and Y are initial observations, and α is the y-intercept.

The parameters are calculated using ordinary least squares (OLS) method as shown in the relation Eq. (2)

$$\beta = \frac{\text{covariance}(X, Y)}{\text{variance}(X)} \tag{2}$$

$$\beta = \frac{\left(\sum(XY) - \sum(X)\sum(Y)/n\right)}{\left(\sum X^2\right) - \left(\sum X\right)^2/n} \tag{3}$$

$$\alpha = \bar{Y} - \bar{X}\beta \tag{4}$$

where $\bar{Y}$ and $\bar{X}$ are the means of X and Y.

But everything has its own set of advantages and limitations so we could not apply linear regression in every case. Application and understanding are very easy and intuitive. A person only with the knowledge of mathematics at high school can understand and apply it easily. In addition to it, it works in most of the cases. Although it does not fit the data exactly, we could at least use it to determine the nature of the situation between the two data. But its disadvantages take over its advantages. Some of them can only model relationship within dependent and non-dependent variable that are linear as it considers always a straight-line relationship which are not true. So, in order to avoid those, we go for a multiple linear regression.

1.4.2 Multiple Linear Regression (MLR)

A multiple linear regression is a method of predicting the outcome of a response variable using several predictor variables [11]. It is an extended version of simple linear regression. It is considered when we want to forecast the value of one dependent variable and two or more independent variables. The resultant variable is called the dependent variable or target variable.

A multiple linear regression model with explanatory variables $X_1, X_2, \ldots X_k$ and a response Y is given below as

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{k-1} X_{k-1} + \beta_k X_k \tag{5}$$

In some cases, a dependent variable is not expected to be described by just one variable. User uses a multiple regression in this case to describe a predictive variable who use more than one independent variable. In the case of linear regression, multiple regressions can be linear or nonlinear.

A multiple linear regression has several techniques to build an effective model such as backward elimination and forward selection and bidirectional elimination. Here, only focus is on backward elimination technique to create the model.

2 Proposed Work

Here, a multiple linear regression model with backward elimination method is used to predict the closing price of Tata Consultancy Services stock index. The data set of TCS stock market has been considered for 5 years in the model from October 28, 2014 to October 25, 2019. The input given to the model is open, high and low (independent variables) and the output of the model is the close price of the dataset.

2.1 Backward Elimination

Backward elimination will help us build our optimal model. Here, all the independent variables are put in the equation and their p-value is checked for deletion of the variables that do not help in building a better model. Before explaining the backward elimination process, the significance level is considered to be 0.05 which determines the statistical hypothesis to be accepted or to be rejected. Our hypothesis states that the difference between actual and predicted value is nearly equal to 0.

In multiple linear regressions, the model does not accept the constant β_0 as in the.csv file these is no constant. But we can take X_0 as one and multiply it to β_0 so that it will be accepted by our model. So a column of one is added to the matrix of features (matrix of independent variables).

Steps to be followed:

Step 1. Select a significance level to remain in the model.
Step 2. Fit the forecast model with all possible cases of independent variables.
Step 3. Consider the independent variable with highest p-value.
Step 4. If p-value is more than the significance level delete the independent variable, fit the model and repeat Step 3.

If p-value is minimum than the selected significance level your model is ready.

2.2 Why We Go for Backward Elimination Method?

While importing data to the model, we should focus on selection and creation of variables based on feature extraction. Feature scaling in machine learning begins with an initial set of observed data and helps in building of the dependent variable.

The figure describes how initially the input data set is fed to backward elimination method and then split validation is performed to select the best inputs to the prediction model (Fig. 1).

In feature scaling, two steps will be carried out feature selection and feature creation.

Here, feature selection is used to study the relation between variables and feature creation and is used to create or select only those independent variables that have a strong co-relation factor.

Advantages of using backward elimination:

1. The training time reduces rapidly.
2. Complex mapping into linear mapping.
3. Avoids over fitting.

Now, we have the *R*-square intuition. It is the goodness of fit. The closer to one it is the better the model. *R*-square will never decrease if we add new variables. It will either remain same by putting the new variable, i.e., the value of the new variable will be zero if it makes the model worse or it will increase by giving a new coefficient that will help minimize the sum of square of residuals.

In Table 1, all the p-value is same, i.e., 0 which is less than our significance level, i.e., 0.05. So, we can say that all the independent variables are equally important for prediction our dependent variable.

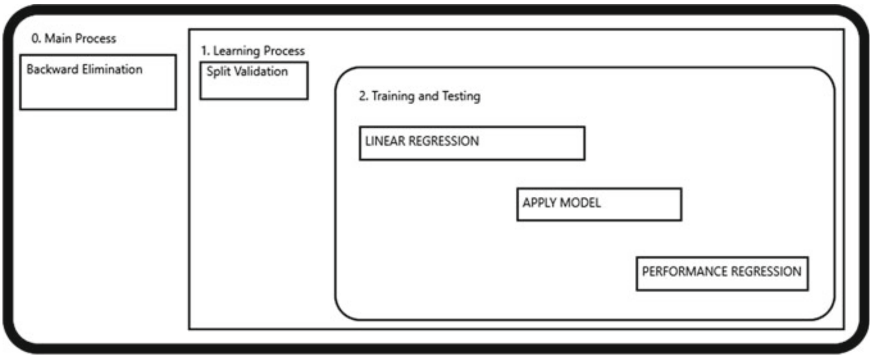


Fig. 1 Flow process of the proposed model

Table 1 Selection of variables using p significance value

	Coefficient	Standard error	t	$P > \|t\|$	[0.025	0.975]
const	−1.2547	0.946	−1.326	0.185	−3.110	0.601
X1	−0.5769	0.018	−31.350	0.000	−0.613	−0.541
X2	0.8156	0.018	44.529	0.000	0.780	0.852
X3	0.7620	0.018	41.775	0.000	0.726	0.798

Table 2 OLS regression analysis of the data set

OLS regression results				
Dep. variable	Y		R-squared	1.000
Model	OLS		Adj. R-squared	1.000
Method	Least squares		F-statistic	8.779e + 05
No. of observations	1229		Prob (F-statistic)	0.00
Df residuals	1225		Log-Likelihood	−4267.9
Df model	3		AIC	8544.
Covariance type	Non-robust		BIC	8544.
Omnibus	119.077		Durbin-Watson	2.063
Prob (Omnibus)	0.000		Jarque-Bera (JB)	821.721
Skew	0.008		Prob(JB)	3.68e-179
Kurtosis	7.006		Cond. No.	1.13e + 04

2.3 Result and Discussion

The data set is the stock price of TCS stock index for the last five years are used as input to the model. The below table is obtained using OLS regression using backward elimination method. Moreover, in Table 2, the R-square is exactly 1. So, we can say that the independent variables are strongly co-related and the regression prediction perfectly fits the data taken in the model. The principle of a multiple linear regression is to evaluate whether a continuous dependent variable from a set of independent (or predictor) variables can be predicted. Omnibus–D'Angostino's test shows a combined statistical test for the occurrence of skewness and kurtosis. The lower value of skewness shows the symmetry of data points around the mean. The higher value of kurtosis shows the large distribution of the data points around the mean (Fig. 2).

Here, in the graph above, we have taken opening price as x and closing price as y and constructed the linear regression graph. It shows the closeness between the actual closing price and the predicted closing price. The graph in Fig. 3 shows the error graph that represents the difference between the actual and predicted closing price.

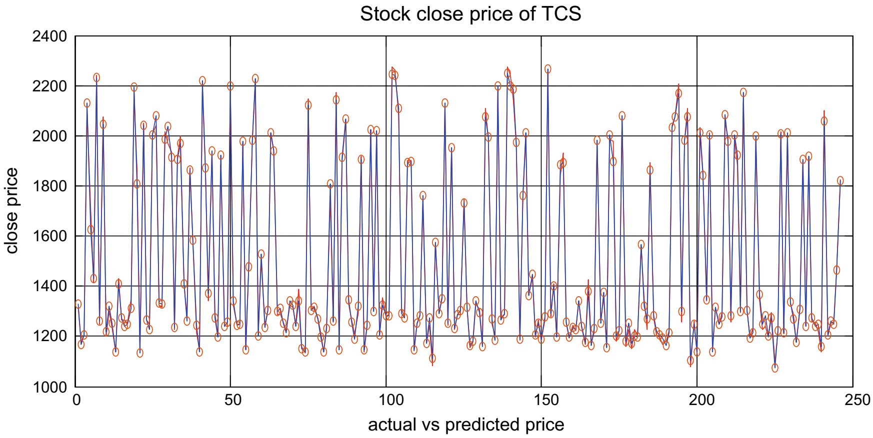


Fig. 2 Actual close price verses predicted close price of TCS stock index using a multiple linear regression

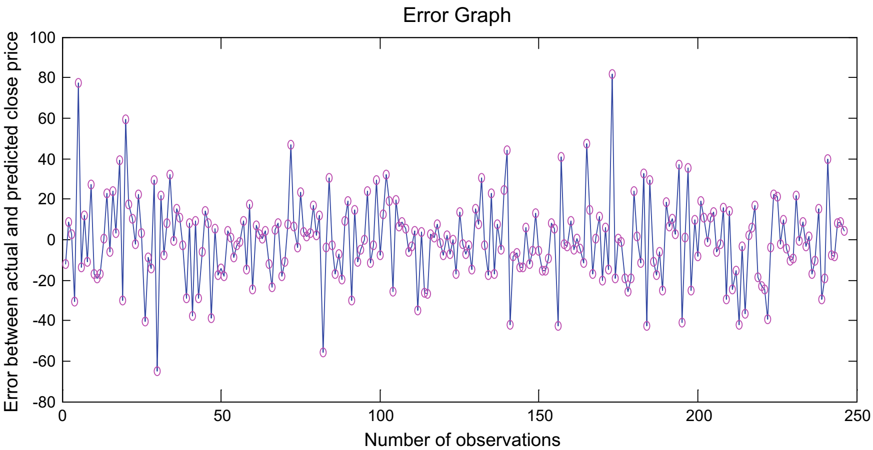


Fig. 3 Error between actual and predicted close price

3 Conclusion

Here, a multiple linear regression model with backward elimination technique is used to predict the TCS stock index. In addition, regression analysis was chosen from the comprehensive research on the nature and the origin of the data being analyzed as it provides criteria of relationships between the predictor and the variables of response. Therefore, regression analysis was chosen from comprehensive research on the nature and source of the data being analyzed as it offers objective measurements of the relationships between the indicator and the response variables. However, regression analysis was selected from comprehensive research on the nature and origin of the data being analyzed as it provides objective measures. The resulting p-value was identified as the predictors used to create the response variables for predictors that were less than 0.05.

References

1. Forslund, G., Åkesson, D.: Predicting share price by using multiple linear regression (2013)
2. Seethalakshmi, R.: Analysis of stock market predictor variables using linear regression. Int. J. Pure Appl. Math. **119**(15), 369–378 (2018)
3. Kumari, K., Yadav, S.: Linear regression analysis study. J. Pract. Cardiovasc. Sci. **4**(1), 33 (2018)
4. Nguyen, D., Smith, N. A., Rosé, C. P.: Author age prediction from text using linear regression. In: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. Association for Computational Linguistics, pp. 115–123 (2011)
5. Mekparyup, J., Saithanu, K., Buaphan, M.: Multiple linear regression analysis for estimation of nitrogen oxides in rayong. Glob. J. Pure Appl. Math. **10**(5), 769–774 (2014)
6. Salleh, F. H. M., Zainudin, S., Arif, S. M.: Multiple linear regression for reconstruction of gene regulatory networks in solving cascade error problems. Adv. Bioinform. (2017)
7. Suparta, W., Putro, W. S.: Using multiple linear regression model to estimate thunderstorm activity. In: IOP Conference Series: Materials Science and Engineering, **185**(1), 012–023 (2017)
8. Altay, E., Satman, M.H.: Stock market forecasting: Artificial neural network and linear regression comparison in an emerging market. J. Financ. Manag. Anal. **18**(2), 18 (2005)
9. Ichise, M., Toyama, H., Innis, R.B., Carson, R.E.: Strategies to improve neuroreceptor parameter estimation by linear regression analysis. J. Cereb. Blood Flow Metab.Ism **22**(10), 1271–1281 (2002)
10. Bianco, V., Manca, O., Nardini, S.: Electricity consumption forecasting in Italy using linear regression models. Energy **34**(9), 1413–1421 (2009)
11. Shakhla, S., Shah, B., Shah, N., Unadkat, V., Kanani, P.: Stock price trend prediction using multiple linear regression. Int. J. Eng. Sci. Inven. (IJESI) **7**(10), 29–33 (2018)

Cloud, Edge and Fog Computing in Healthcare

Bichitrananda Patra and Karisma Mohapatra

Abstract Cloud, Edge, and Fog computing be present one of the productive approaches in which system shares among plans and cloud revelation off application-definite sense. This product permits a survey of the fog and mist computing in health-care informatics, classifying them into complementary collections and inspects about the, unlike presentations of the method through this paper. In errand, to that, the solicitations are categorized into use-instance classes and are planned in indexing of application-definite fact that can be directed by fog and edge computing. It is added in advance that on which layer of the network system these fog and edge computing tasks can be measured and trade-offs about the health care requirements are made. It is vindicated by quoting altered life situations that: in the present times, in peace of which, a record of recognized analysis and growth performance are offered.

Keywords Fog computing · Edge computing · Cloud computing · IoT

1 Introduction

The near up-coming developments and research in the technology have adjustment the epicenter of the medical sciences in rapidly not only for analysis but rather how to prevent those diseases with incomplete recognition and appropriate and precise health information through leading in-demand technology called the "Internet of Things." In order to record physiological symptoms and signals, various efficient devices have been discovered and are feasible in the market which can be easily connected to the internet over smartphones, computers, or any nodal devices. There is a high potential appeal of computing the fog inside the IoT-based monitoring systems as

B. Patra (✉)
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India
e-mail: bichitranandapatra@soa.ac.in

K. Mohapatra
Department of Computer Science, Christ College, Cuttack, India
e-mail: mohapatra.karisma@yahoo.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_57

per the contemporary surveys. The IoT states to the emergent system of somatic substances that scope an IP address for network connectivity with unique identities are interconnected for enabling the various objects to interact and act accordingly. For obtaining various essential signs within real-time, efficient and working in the active way occurring like ECG, EMG, BP, WBAN (Wireless Body Area Networks), and body temperature are used as among the basic technologies in IoT methodology of healthcare.

WBAN establishes a multitude of impalpable or attire sensor nodes to percept and conveys the data over a wireless network. The transmission protocols used for this are Wi-Fi or IEEE.802.15.4. Basically, WBAN-based structure is Low-cost and power competent and plays a leading role in several regions of healthcare environments such as tracking, medical care to chronic virus controlling, and disease precaution. Distant cloud servers are used in many health monitoring systems for keeping and organizing huge data possessed from a huge quantity of sensor nodes. Cloud computing has a number of benefits like a small budget amenity, capacity of huge records stowage size, and surplus preservation cost but few challenges also exist like large data transmission, location perception, and latency-sensitive issues. Due to the slowdown in data transmission and data error packet dropping possibilities have increased. More the data are sent over a network; greater is the chance of mistakes that may result in wrong or less precise treatment decisions due to which there will be adverse effects on emergency situations and critical illness of mankind. Thus, there will be a great requirement of minimizing the flow of data absorbed ended one net with the quality of services (QoS). One solution for accomplishing this need is the implanting of an additional coating in the middle of a straight gateway and a distant cloud server. This added layer is termed as fog layer ad it aids in lowering the large chunks data and yet guaranteeing QoS, so redeemable system bandwidth by pre-analyzing the data. In accumulation, fog computing delivers innovative amenities at the control of the net then decreases the load of cloud [1]. Although the cloud computing pattern is brought to the edge of the net by cloud computing, it also reports unproven or frail basics in the cloud computing pattern. Certain of the essential characteristics of fog computing are low latency, geographical dissemination, edge localization, locality awareness, interoperability, and assist for connected analytics. That is why fog computing can be taken into consideration for advancing the human health observing WBAN-based systems due to the feature like less bandwidth, less energy, less processing control and involve hardware inhibited knobs. So say in literal words, a mishmash of WBAN-based system, cloud, and fog computing may be a defensible result for summoning in the present IoT healthcare system. At here efficient IoT-enable healthcare structure planning is presented which is profited from the idea of fog and edge computing. By this arrangement, we come up with effective fog computing and edge computing healthcare systems which will be IoT enabled and bandwidth utilization, QoS guarantee, and emergency reminder. Briefly, we can quote the following features discussed here.

- Healthcare IoT-based system need, along with benefits given by Fog computing layer is addressed and elaborated here.

- A system of IoT and Fog-enabled healthcare is shown here with its architecture as well.
- Proof-of-concept full system implementation of the fog computing layer services with performance and advantages are shown.

2 Related Work

There have already been many attempts in creating brilliant access aimed at healthcare function. For example, Chen et al. [2] introduced a wireless sensor network that is used as a keen gateway for the fitness care structure. The endorsed gateway operating as an overpasses amongst wireless sensor net and community nets. Mohapatra et al. proposed semi hybrid architecture for distant long-suffering observing which uses a sensor cloud [3]. Advantage of utilizing a sensor cloud for patient's fitness form observing is shown in their projected system. The writers [4] did present a cloud computing clarification for patient's records gathering in healthcare organizations. Sensors are used in the system which is adhered to medical equipment to assemble patient records and sent the records to the cloud to provide restricted admittance. Yang et al. presented individual health observing access constructed on smartphones [5]. The recommended gateway requires a Bluetooth console to upload collected records to distant servers. An assessment table is given in Table 1 to compare the features between cloud and fog computing. This project is shown in [6], these sensing servers are used as gateways in the system which is handled by the sensor network system. But, the proposal is exaggerated the insufficient, and extensible as well as inefficient for many applications. In [7], authors have researched and proposed an idea of an exemplar of a lively IPv6. Low power wireless area network (6LoWPAN) a router that was constructed on Hidden Markov Model. This model was used for building a settlement of fitness status. In [8], the writers have created a movable gateway for pervasive healthcare system using ZigBee and Bluetooth. The gateway

Table 1 Assessment of some features of cloud and fog computing

Features	Cloud computing	Fog computing
Node place	The Internet	Local network edge
Quantity of node	Little	Large
Latency	High	Low
Delay	High	Low
Distance client to server	Multiple hops	Single hop
Locality alertness	No	Yes
Delivery	Centralized	Distributed
Flexibility support	Limited	Good
Data storing	Huge	Limit
Communication	Device to cloud	Device to device

favors to feebleness many services such as investigation of health records. However, contemplate the gateway is ineffective power utilization and it cannot be measured for practical use. Zhong et al. presented an alternative way created over a mobile linking sensor network knobs and devices promoting CDMA or Bluetooth [9]. One more slog [10], these expected plans reach records through many individual health strategies. The communication mode is considered as USB and Bluetooth, etc. The subject of body part sensor network structures has pinched the attention of more researchers in the modern years. Specifically, the numeral of mechanisms in the healthcare field has exponentially expanded to various undiscovered stepping stones with only one intention for the betterment of health monitoring systems.

3 Fog Computing

Fog computing, networking, storage, and other domain with particular services are handed over to the IoT system by Fog computing layer. The healthcare domain differentiates it from the other IoT applications by having the most used feature of remote monitoring which requirements a high degree of reliability. The safety and secrecy aspects of health care are highly important and should be implemented along with the Fog layer through an emphasis on health care. Some of the healthcare IoT amenities are inclusive then may be used by any presentation domain. The concluded view of the Fog layer services is deliberated distinctly and shown on Fig. 1.

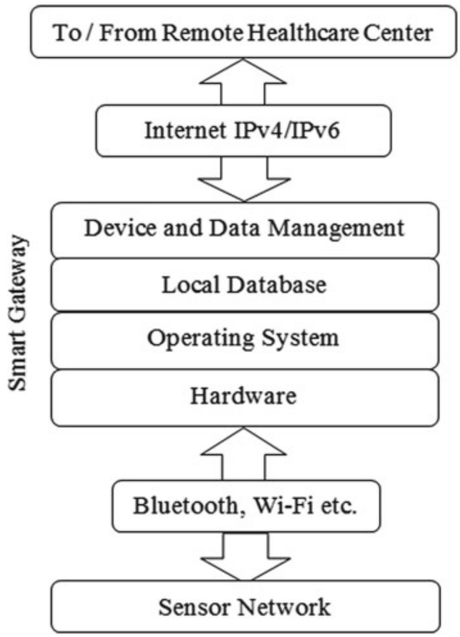


Fig. 1 Different Fog layer and services

3.1 Managing Data

Data management plays a main part in Fog computing through which the sensual data is nearby used in order to withdraw essential information for anyone to provide a response to the user and announcements along with the system design adaptations. Giving to the system architectonics, fog layer dynamically accepts a huge quantity of auricular facts in a very small intermission of time from the sensor net, thus its main focus is to control the incoming data in order to give a fast reaction with respect to several user and system environments. This mission acquires further notable in healthcare states subsequently potential and disquiet tend to happen during decision-making and may cause incurable damages to the patient. Permitting to the numerous fulfillment of the data supervision work, we established it with altogether five dissimilar things, entirely of which are requirements in an intelligent e-health gateway.

3.2 Managing Event

A number of major events take place in patient audit some being an alteration in vigorous symbols, deeds, or surroundings of the patient. Every event activates a specification of gateway before shifts in an erudite behavior on actualize system. Fog computing supplies less potential clarification which supports advising health connoisseurs, caretaker. A number of major events take place in patient audit some being an alteration in vigorous symbols, deeds, or surroundings of the patient. Every event activates a specification of gateway before shifts in an erudite behavior on actualize system. Fog computing supplies less potential clarification which supports to advise health connoisseurs, caretaker besides equal patients need to be rapid in case of critical condition. In such situations soon as a quick reaction is needed in case of medical activities or unconscious system actuation, the occurrence administration facility fortifies on time and appropriate signal transport. The actual with the quick reaction of activators is necessary for certain medicinal procedures alike shifting the frequency of nerve stimulation rendering to emotion speed or regulating the unconscious insulin impel through glucose level. Additional mandatory incidents potency also occurs to which is requirements to be alerted to the rapid responding team, caretakers, or family associates of the patient.

3.3 Source Effectiveness

In healthcare IoT function, source proficiency is some of the greatest important factors as catastrophe in resource management can lead to the simple corollary due to device nodes' failure to incorrect disease analysis. Specifically, liveliness exhaustion of device nodes and expectancy of collected data offered at end-users terminals essential to be constantly kept a note of. They are dealt with as follows:

- Efficient use of energy of the nodes
- Latency

3.4 Management of System

Device managing consists of many parts of Internet of Things organization. In this part, the main area of the debate is on a device supervising from the fact of observation of locating the device and keeping the connection through in mobility.

3.4.1 Locating and Flexibility

Before the reserve restricted in the sensor and actuator systems have been stated. The lifetimes of cells are very important and actual supervision requires. Devices must be in sleep mode in a supervised manner whenever they are about to gain idle condition. Any transmission that takes place when the device is in the sleep condition requires looking care of through the Fog cover. In the healthcare scenario, one persistent trying medicinal sensors and affecting since a position to other adapts the liable gateway the changes transmission. This shows that the sleeping sensors essential to waken up at the sector of various gateways. A full utility of this provision can be found [11].

3.4.2 Interoperability

The IoT is generally a bunch of diverse usual transmission protocols and data setups on a platform. There are multiple regularities that try to initiate constantly amongst the various modules. The present upright subdivisions of presentations require to be joined to simplify the modeling of not, just a part of but the entire of the healthcare applications. Traditionally affinity is facing the dare related to resource restriction of most of the end devices. The Fog figuring layer caters a vital part in giving amenities that grip the presence of its cover to the last device, thus affluence interoperability. These efficacies work as a connector between several communication protocols, data setups, and platforms.

3.5 Personification

Machine behavior can be set up for various presentations of Fog computing now trial or through in ride time. This although force is not adequate aimed at healthcare structure since users might have a numerable medicinal state besides hold in various stringers in assorted surroundings. So, a powerful plan for the machine is necessary, not only to customize the machine performance in granting to specific consumer desires but also assistive habituation to the machine over time, exceptionally in case of a crisis. In this regard, it enhances the local decision-making as well as optimizes the energy proficiency. Customized scheme behavior may be illustrated of several health solicitations by statute created skills and machine learning processes. By these conclusions, many primaries and means are described as per the device specimen frequency and data diffusion ratio, and in accordance with the patient's condition, the proper principles are chosen. Further, the precedence becomes customized with respect to the medical history of the patient. Simply put together, suppose heart failure is diagnosed for persevering through in the observation, the machine would acquire to give more proper heart associated frameworks.

3.6 Privacy and Security

Overall, safekeeping is very vital for every solicitation and stands further crucial in circumstances of healthcare due to an only apprehensive point in a machine that might cost a social lifespan. An insulin pump in IoT glucose adjusting a mechanism may persist scythed within 100 feet [12]. To deliver a protected IoT healthcare machine, the full machine which consists of device nodes, gateways, Fog, and Cloud essential be protected adequately. If any components get hacked, the complete system can easily be operated by hackers. Designed for instance, various means such as CMAC-AES-128 or AES-128 might be used at device knobs besides gateways for facts encryption and decryption correspondingly. Linux provides IP tables that may be utilized for constructing IP table presenting consent to proper transmission docks. However, this technique can modify safety at different levels, it can't be trusted as a powerful technique for securing the whole machine. Whereas furthers occasions in the literature [13–15] afford an extraordinary level of security. Still, they can't be concerned in IoT machines since these complicated cryptographic procedures are not concern able to source-constrained sensor nodes. Toward directing the safekeeping troubles in IoT healthcare machine, Rahmani et al. [16, 17] currently launch end-to-end protected parameters. The parameters may give better verification and approval for the Health Internet of Things machine while the chief parts, comprising of various complicated security algorithms, are executed at the Fog layer.

4 Edge Computing

An edge computing says about an astonishing eye opening technology that allows calculating at the extremity of the linkage, on data of downstream or upstream kind. Downstream will come from cloud services whereas upstream data is organized on the part of IoT services. And now we declare "edge" as figuring and integrating network system locality along the way among data obtain ability and cloud data servers. Like, a smartphone is a border or edge connecting living beings and cloud data, a gateway can be referred to as a smart home. The hypothesis of the edge computing is that computation should always take place near a data source. According to our idea of research, fog computing is replaced by edge computing [18], however, edge computing stress mainly on the effects adjacent, whereas fog computing mark extra on the substructure adjacent. It envisages that edge computing might possess as large an effect in our community just like the cloud computing. Display in Fig. 2 that the mutual computing rush in edge computing. Edge computing models thing are not only acts as data consumers but also act as the data producers. On the edge, the possessions may not only appeal facility and content from the cloud but also represents the computing works commencing on the cloud. An edge can conduct computing transfer, data storage, accumulating and handling, as well as spread appeal and deliver the amenity from cloud to user. With these kinds of tasks in the network, the edge itself requires to be well planned to converge the need effectively with praise to consistency, security, and isolation shield.

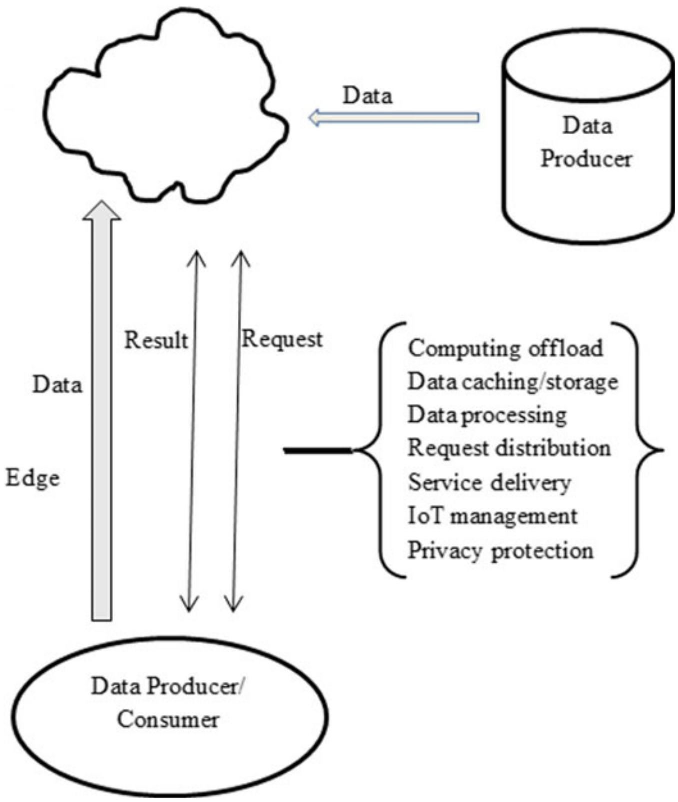


Fig. 2 Edge computing architecture

Now in edge computing, the center of attention is on setting the computing attached to the closeness of data causes. This has various advantages in contrast to conventional cloud-form computing patterns. At here we are using several premature outcomes from the circle to illustrate the probable advantage. Research workers elevated a proof-of-concept stand to execute face response appearance [18] and the reaction period is basically condensed from 900 to 169 ms by just poignant computation from cloud to edge. Ha et al. [19] recycled cloud let's unload the computing jobs for wearable reasoning support and the end result shows that the advancement of reaction time is among 80 and 200 ms. Furthermore, the energy utilization might be condensed by 30–40% it means of cloudlet offloading. Clone cloud merge dividing, relocation with integration, and the on-demand instantiation of dividing between the mobile and the cloud. And their prototype could shrink 20 times the executing period and energy for verified uses.

5 System Architecture, Fog, and Edge Computing

Figure 3 displays a comprehensive figure upon in what way the rudiments of a Health IoT structure can be systemized in a discrete technique finished the three sheets to be recycled in keen hospitals. Now, these groupings, patient health connected substantial is renowned by entrenched sensors, thus helping the patient furnished with private monitoring of various variables. This health data can be also improved by means of the date, time, location, temperature, etc. Circumstance-awareness allows identifying a typical project and makes more exact implications regarding the situations. Other medicinal kits can also be devoted to the machine to transfer information to the staff

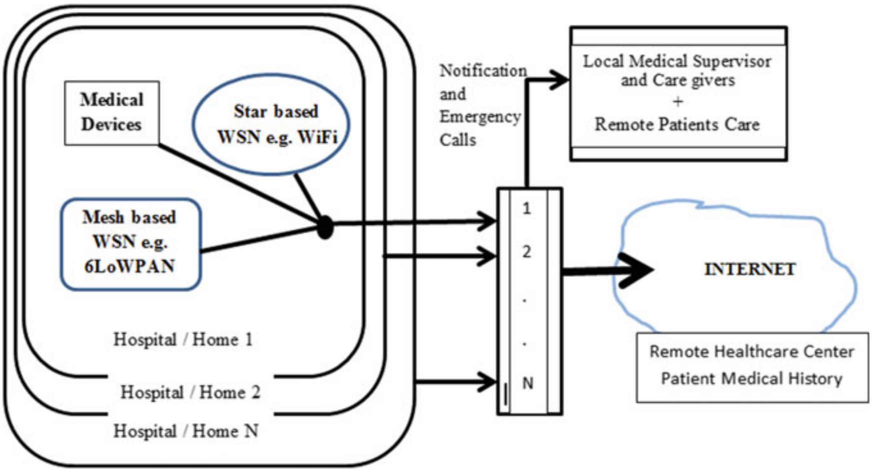


Fig. 3 Smart e-health monitoring system

like CAT examination, captivating quality imaging, etc. The machine construction embraces the succeeding main elements.

- Medicinal Sensors and Actuators Network: These are allowed through omnipresent documentation, detecting, and transmission competence, biomedical and perspective signals remain taken on or after the body and neighboring. Statistics is former lyre located to the access via wireless or supported dispatch rules that is Bluetooth, Wi-Fi, Net of Intelligent e-health.
- Gateways: This level is constructed on or after several physically scattered intelligent e-health gateways, such as developing a virtual fog. Every gateway, which helps altered transmission rules, acts as a vigorous connection idea among a sensor network and the local switch or Internet. It accepts data from various subnetworks, bearings protocol adaptation, and gives further high-level facilities like data accumulation, data filtering, and dimensionally depletion.
- Back-End System: This system comprises a cloud computing programme that performs streaming, data depository as well as data analysis. Lastly, the situation adjunct the web client exhibition just as GUI regarding final picture and criticism. The possessed health and setting data to constitute a source of large data [20, 21] for arithmetical and epidemiological detecting forthcoming.

5.1 Properties in Addition to Features of Smart e-Health Gateways

As explained before, the important share of a gateway is to help several wireless proprieties besides gross charge of inter-device transmission. In this part, we enlarge its part turn out to be fog enable by (1) establishing a composed linkage of gateway (2) applying several features like acting as repository to volatile stock beams' and users' data and combining it with data merging, collection and clarification methods. These are vital to give native pre-dispensation of sensors' records, consequently may be called an intelligent e-Health Gateway.

6 Conclusion

The research work presented in this article focuses on retrospection of Edge and Fog computing applications in healthcare. These jargons are associated with cloud computing and named according to their architectural relationship. Various applications can function more effectively using Edge and Fog level rather than migrating to the cloud. This is due to various factors like the nature of client, data locality, network overhead, device and cloud resources, their availability, etc. The main intention remains to make the service available in all types of circumstances. It has been found that various healthcare-related applications are more applicable for execution in Edge and Fog rather than cloud due to its solution to the various constraints of the various sensors.

References

1. Burt, J.: Fog computing aims to reduce processing burden of cloud systems. http://www.eweek.com/networking/fog-computing-aims-to-reduce-processing-burden-of-cloud-systems.html (2010)
2. Chen, Y., Shen, W.: Huo, H., Xu, Y.: A smart gateway for health care system using wireless sensor network. In: 2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM), July, pp. 545–550. (2010)
3. Mohapatra, S., Rekha, K.S.: Sensor-cloud: a hybrid framework for remote patient monitoring. Int. J. Comput. Appl **55**(2), 1–11 (2012)
4. Rolim, C.O., Koch, F.L., Westphall, C.B., Werner, J., Fracalossi, A., Salvador, G.S.: A cloud computing solution for patient's data collection in health care institutions, in ETELEMED'10. In: Second International Conference on eHealth, Telemedicine, and Social Medicine, pp. 95–99. IEEE (2010)
5. Yang, S., Gerla, M., Personal gateway in mobile health monitoring. In: 2011 IEEE International Conference on Pervasive Computing and Communications Workshops PERCOM Workshops, pp. 636–641, March 2011
6. Yoon, J.W., Ku, Y.K., Nam, C.S., Shin, D.R.: Sensor network middleware for distributed and heterogeneous environments, in NISS'09. In: International Conference on New Trends in Information and Service Science, pp. 979–982. IEEE (2009)
7. Shen, W., Xu, Y., Xie, D., Zhang, T., Johansson, A.: Smart border routers for ehealthcare wireless sensor networks. In: 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), pp. 1–4, Sept 2011
8. Laine, T.H., Lee, C., Suk, H.: Mobile gateway for ubiquitous healthcare system using ZigBee and bluetooth. In: 2014 Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 139–145. IEEE (2014)
9. A Phone-Centered Body Sensor Network Platform: Cost, Energy Efficiency & User Interface. Institute of Electrical and Electronics Engineers, Inc. http://research.microsoft.com/apps/pubs/default.aspx?id=103003 (2006)
10. Park K., Pak, J.: An integrated gateway for various Ph.Ds in UHealth care environments. BioMed. Res. Int. **2012** (2012)
11. Negash, B., Rahmani, A.M., Westerlund, T., Liljeberg, P., Tenhunen, H.: Lisa: light weight internet of things service bus architecture. Procedia Comput. Sci. **52**, 436–443 (2015). In: The 6th International Conference on Ambient Systems, Networks and Technologies (ANT-2015), The 5th International Conference on Sustainable Energy Information Technology (SEIT-2015)
12. Paul, N., Kohno, T., Klonoff, D.C.: A review of the security of insulin pump in fusion systems. J. Diabetes Sci. Technol. **5**(6), 1557–1562 (2011)
13. Kambourakiset, G., Klaoudatou, E., Gritzalis, S.: Securing medical sensor environments: the code blue framework case. In Proceeding of the Second International Conference on Availability, Reliability and Security, pp. 637–643 (2007)
14. Chakravorty, R.: A programmable service architecture for mobile medical care. In: Proceeding of Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, pp. 5–536 (2006)
15. Ko, J., Lim, J.H., Chen, Y., Musvaloiu-E, R., Terzis, A., Masson, G.M., Gao, T., Destler, W., Selavo, L., Dutton, R.P.: MEDiSN: Medical emergency detection in sensor networks. ACM Trans. Embed. Comput. Syst. **10**(1), 11:1–11:29 (2010)
16. Moosavi, S.R., Gia, T.N., Rahmani, A., Nigussie, E., Virtanen, S., Tenhunen, H., Isoaho, J.: SEA: a secure and efficient authentication and authorization architecture for IoT-based

healthcare using smart gateways. In: Proceeding of 6th International Conference on Ambient Systems, Networks and Technologies, pp. 452–459 (2015)
17. Moosavi, S.R., Gia, T.N., Nigussie, E., Rahmani, A., Virtanen, S., Tenhunen, H., Isoaho, J.: Session resumption-based end-to-end security for healthcare Internet-of-Things. In: Proceeding of IEEE International Conference on Computer and Information Technology, pp. 581–588 (2015)
18. Yi, S., Hao, Z., Qin, Z., Li, Q.: Fog computing: Platform and applications. In: Proceedings of 3rd IEEE Workshop Hot Topics Web Systems and Technologies (HotWeb), Washington, DC, USA, pp. 73–78 (2015)
19. Ha, K., et al.: Towards wearable cognitive assistance. In: Proceedings of 12th Annual International Conference on Mobile Systems, Applications, and Services, Bretton Woods, NH, USA, pp. 68–81 (2014)
20. Laney, D.: 3D data management: controlling data volume, velocity, and variety. Technical report, application delivery strategies by META Group Inc. (2001)
21. Beyer, M.: Gartner says solving 'Big Data' challenge involves more than just managing volumes of data. http://www.gartner.com/newsroom/id/. 1731916 (2016)

An Empirical Comparative Analysis of NAV Forecasting Using Machine Learning Techniques

Sarbeswara Hota, Sanjaya Kumar Jena, Bhupendra Kumar Gupta, and Debahuti Mishra

Abstract Mutual funds become the mode of investment for the common people. Net asset value (NAV) of a mutual fund is one of the performance indicators. The NAV data are nonlinear in nature and form the financial time series data. So machine learning methods are useful in developing forecasting models. In this paper, different variants of neural network models, i.e., multilayered perceptron (MLP), extreme learning machine (ELM), and functional link artificial neural network (FLANN), are used for the 1-day, 3-day, 7-day, and 15-day ahead NAV forecasting of one of the Indian mutual funds. The NAV data are divided into training and testing data with 8:2 ratio. The performance of these models is evaluated using RMSE and MAPE values. The experimental results demonstrate that ELM outperformed the other two models in forecasting the NAV values for these two Indian mutual funds.

Keywords Net asset value · Extreme learning machine · Functional link artificial neural network · Backpropagation · Mean square error

1 Introduction

In the current financial market, mutual funds are the most common investment tool [1]. The mutual fund investors should know the net asset value (NAV) of the trade date

S. Hota (✉)
Department of Computer Application, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha 751030, India
e-mail: sarbeswarahota@soa.ac.in

S. K. Jena · B. K. Gupta · D. Mishra
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha 751030, India
e-mail: sanjayjena@soa.ac.in

B. K. Gupta
e-mail: bhupendragupta@soa.ac.in

D. Mishra
e-mail: debahutimishra@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_58

of different funds, and it helps in evaluating the future performance of those funds before putting their money. It is not possible to forecast the mutual fund performances for the common investors and financial advisors, particularly NAV prediction of mutual funds. So NAV prediction has become an important research domain [2]. Different statistical models have been developed to forecast the financial time series data. Most of these statistical models assume that the dataset is generated from a linear process [3, 4]. However, the current financial time series data are dynamic, chaotic, and highly nonlinear. The most important aspect of forecasting models is to handle the incomplete and irregular data. Machine learning techniques have found to be useful in handling these data. So different machine learning techniques have been applied in developing the financial time series forecasting task and found to outperform the other statistical models [5].

From the literatures, it is found that many financial time series data follow nonlinear behavior, and the statistical methods are not sufficient to reflect the dynamics. So the different neural network models have been widely accepted by the research community for the development of time series forecasting model. In [6], the authors used backpropagation neural network (BPNN) model for the prediction of net asset value of 101 US mutual funds. The network parameters were optimized experimentally. The authors identified 15 economic variables as input to the neural network model. It was found that the backpropagation neural network model performed better than the econometric models. However, this was a very basic neural network model, and many powerful extensions would have been developed for time series forecasting models. Moghaddam et al. in [7] performed the daily NASDAQ stock exchange rate prediction using different structures of multilayer perceptron. The authors found the optimized ANN with three hidden layers and 20-40-20 hidden neurons and used backpropagation algorithm for training the network. The authors used R^2 value as the performance measure in this study. Gurusen et al. used the multilayer perceptron (MLP), dynamic ANN model, and hybrid ANN model using generalized autoregressive conditional heteroscedasticity (GARCH) for stock market index prediction [8]. The results obtained using hybrid ANN model were better as compared to other two models. The literature study indicates that different variants of neural network models, i.e., MLP, extreme learning machine (ELM), and FLANN models, are applied in financial time series forecasting. The authors in [9] used the FLANN model to forecast the NAV of different Indian mutual funds. The NAV data were combined with the statistical features to form the input pattern. The input pattern was trigonometrically expanded. For training the FLANN model, 80% of the dataset were taken and for testing purpose, rest 20% of the NAV dataset were considered. The authors compared the prediction performance of FLANN model with MLP and found that the FLANN model was better as compared to MLP model. In [10], the authors applied FLANN model using trigonometric expansion function in predicting the short-term and long-term stock prices. For training the weights of the FLANN model, they used least mean square (LMS) and recursive least square (RLS) algorithms.

The literature study indicates that different variants of neural network models are applied in financial time series forecasting. In this work, the different neural network variants are applied for NAV forecasting of Indian mutual funds. So the goal of this paper is to compare the forecasting performance of MLP, FLANN, and ELM models for the 1-day, 3-day, 7-day, and 15-day ahead NAV prediction of one of the Indian mutual funds. This paper is organized as follows. Section 2 describes the methodologies. Section 3 deals with the simulation study and result analysis. Section 4 concludes the paper.

2 Methodologies

This section describes the ELM model and FLANN model.

2.1 ELM Model

ELM is a suitable learning technique proposed by Huang et al. for single hidden layer feed-forward neural network [11]. ELM assigns the weights between the input layer and hidden layer randomly. The weights between the hidden layer and output layer are determined analytically using Moore–Penrose generalized inverse [12]. The ELM model is shown in Fig. 1.

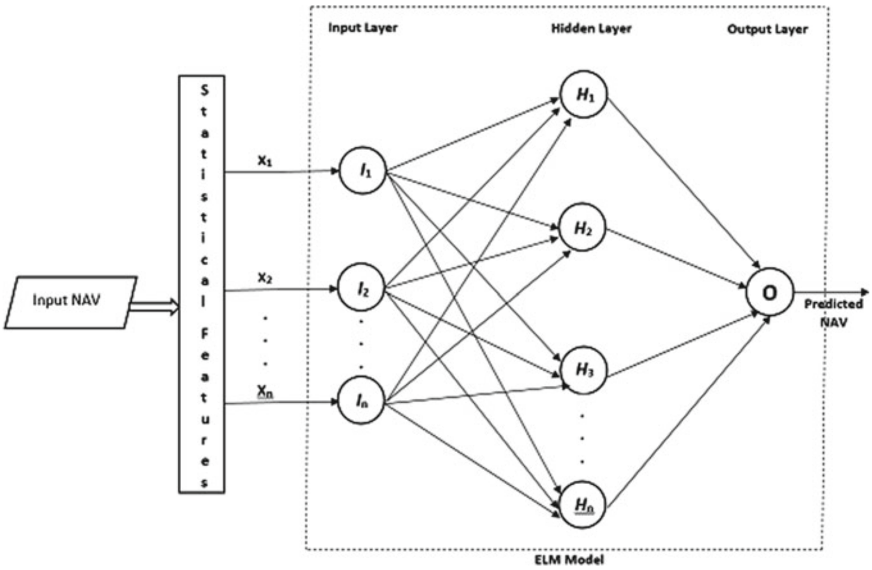


Fig. 1 Basic ELM model

2.2 FLANN Model

FLANN model is a single-layer higher-order neural network model. It consists of an input layer and one output node without any hidden layer. This model is suitable for nonlinear financial time series data prediction task as the functional expansion property provides nonlinearity [13–15]. In this work, the input pattern is expanded through trigonometric expansion to provide nonlinear elements. Various researchers have used trigonometric expansion in FLANN model as expansion function scheme. The output of a node is calculated using the weighted linear combination of the expanded inputs and the activation function. The weighted sum is calculated using Eq. (1).

$$y = f\left(b + \mathrm{W} \cdot X^{\mathrm{T}}\right) \tag{1}$$

where y is the predicted output, b is the bias, W is the weight vector, X is the expanded input vector, and f is the activation function used in the FLANN model. The error is calculated as the difference between actual and predicted output. Using this error value, the weights are updated using LMS algorithm. The block diagram of FLANN model is shown Fig. 2.

In this paper, an attempt has been made to compare the performances between the MLP, ELM, and FLANN model 1-day ahead and 7-day ahead NAV prediction of one of the Indian mutual fund. The proposed working model is shown in Fig. 3.

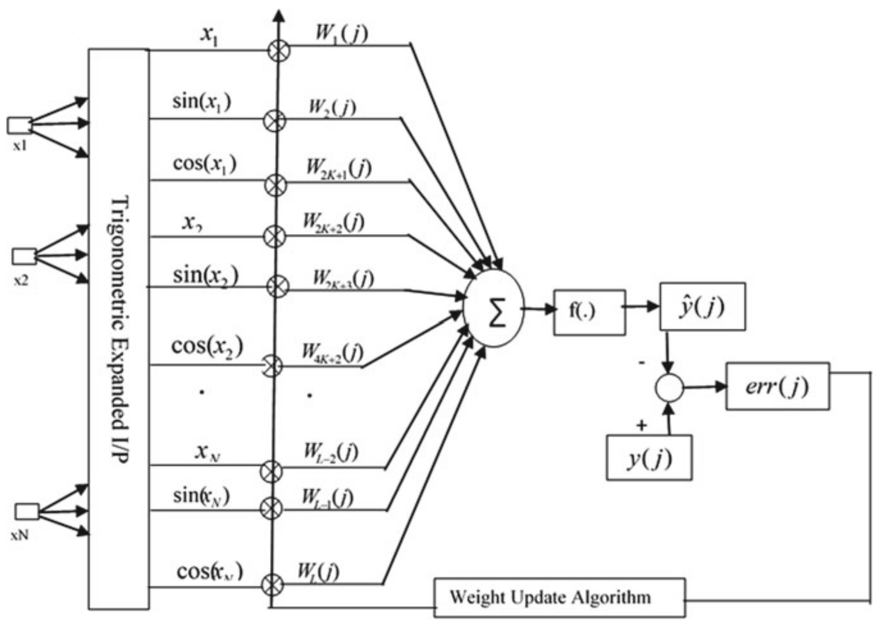


Fig. 2 Block diagram of FLANN

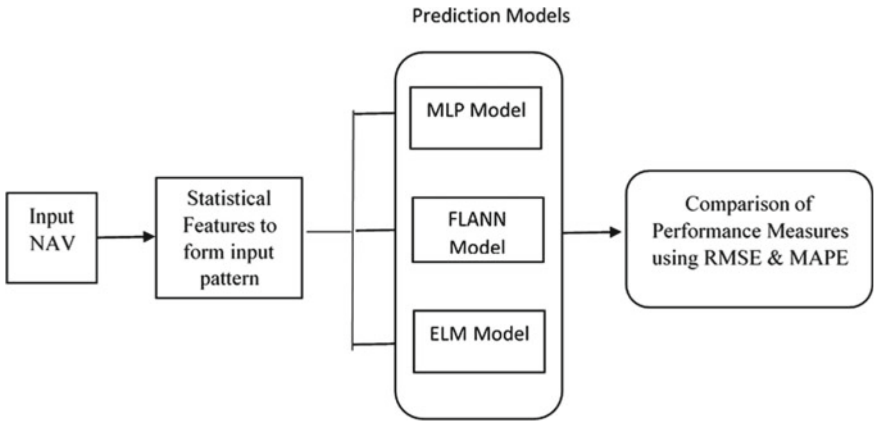


Fig. 3 Proposed working model

3 Empirical Study

The simulation study of this comparison work is discussed in this section. The NAV data of SBI magnum equity mutual fund are collected from March 1, 2007 to March 1, 2017, with 2429 samples. A running window of size 12 is taken to determine the input pattern. For each window, the mean, standard deviation, kurtosis, and skewness are calculated. The input pattern now consists of the current NAV and these four statistical components. The window is moved down by one NAV, the next window is used to decide the second input pattern, and the process continues to prepare the training dataset. In this case, 80% of NAV are taken for training the models, and rest 20% are used for measuring the performance of the model.

The MLP model, ELM model, and FLANN models are trained with the training NAV data. Once the parameters of the models are set, then the models are validated with the test data. The performance of the models is measured using root mean square error (RMSE) and mean absolute percentage error (MAPE). These values are shown in Tables 1 and 2 for the 1-day, 3-day, 7-day, and 15-day ahead predictions.

The actual NAV and the predicted NAV are plotted for the test dataset and shown in Fig. 4 for the 1-day ahead prediction. From the plots, it is found that the two values almost overlap in ELM model.

Table 1 Comparison of RMSE values of different models for SBI magnum equity mutual fund

No. of days ahead	Prediction models		
	MLP model	ELM model	FLANN model
1-day	1.6185	0.00061	1.5663
3-day	1.7654	0.00284	1.6783
7-day	2.1205	0.00315	2.1318
15-day	2.4217	0.00685	2.3369

Table 2 Comparison of MAPE values of different models for SBI magnum equity mutual fund

No. of days ahead	Prediction models		
	MLP model	ELM model	FLANN model
1-day	3.7288	0.005196	3.5943
3-day	4.0564	0.009241	3.8291
7-day	4.8601	0.009846	4.8857
15-day	5.5541	0.062100	5.3397

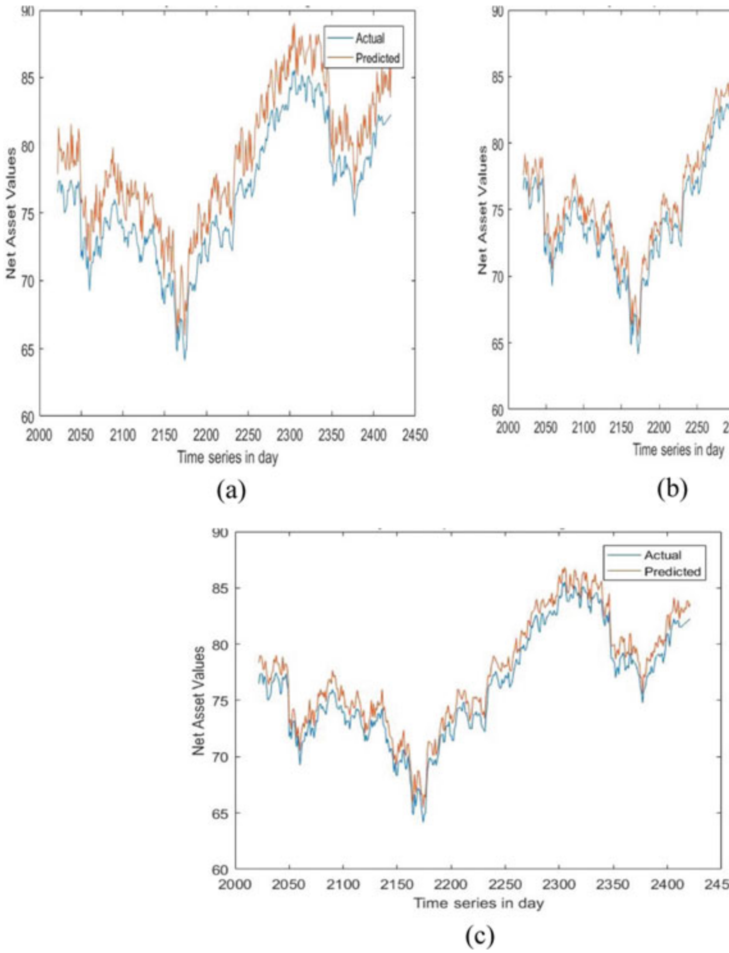


Fig. 4 Actual versus predicted 1-day ahead NAV prediction using **a** MLP, **b** FLANN, and **c** ELM model for SBI magnum equity mutual fund

From Tables 1 and 2, it is found that the RMSE values of ELM model for 1-day, 3-day, 7-day, and 15-day ahead predictions are 0.00061, 0.00284, 0.00315, and 0.00685 respectively. So it is concluded that ELM model performed better as compared to MLP and FLANN models.

4 Conclusion

This paper makes an empirical comparative study between the MLP model, ELM model, and FLANN model for the 1-day, 3-day, 7-day, and 15-day ahead NAV prediction of SBI magnum equity mutual fund. The historical NAV data are collected for a specific period. The data are combined with four statistical features to form the input pattern. The models are trained with the training NAV data, and the prediction performance of these models is validated with the test dataset. The RMSE and MAPE values are taken as the performance metrics. The simulation study indicates that the ELM model outperformed the other two models for the SBI magnum equity mutual fund.

References

1. Cuthbertson, K., Nitzsche, D., Sullivan, N.O.: A review of behavioral and management effects in mutual fund performances. Int. Rev. Financ. Anal. **44**, 162–176 (2016)
2. Gardner, E., McKenzie, E.: Forecasting trends in time series. Manage. Sci. **31**(10), 1237–1246 (1985)
3. Gardner, E.: Exponential smoothing: The state of the art. J. Forecast. **4**(1), 1–28 (1985)
4. Kalra, A., Kumar, S., Waliya, S.S.: ANN training: A survey of classical and soft computing approaches. Int. J. Control Theor. Appl. **9**(34), 715–736 (2016)
5. Henrique, B.M., Sobreiro, V.A., Kimura, H.: Literature review: machine learning techniques applied to financial market prediction. Expert Syst. Appl. **124**, 226–251 (2019)
6. Chiang, W.C., Urban, T.L., Baldridge, G.W.: A neural network approach to mutual fund net asset value forecasting. Omega Int. J. Manag. Sci. **24**(2), 205–215 (1996)
7. Moghaddam, A., Moghaddam, M., Esfandyan, M.: Stock market index prediction using artificial neural network. J. Econ. Finan. Adm. Sci. **21**, 89–93 (2016)
8. Gurusen, E., Kayakutlu, G., Daim, T.: Using artificial neural network models in stock market index prediction. Expert Syst. Appl. **38**, 10389–10397 (2011)
9. Anish, C.M., Majhi, B.: Net asset value prediction using FLANN model. Int. J. Sci. Res. **4**(2), 2222–2227 (2015)
10. Majhi, R., Panda, G., Sahoo, G.: Development and performance evaluation of FLANN based model for forecasting of stock markets. Expert Syst. Appl. **36**, 6800–6808 (2009)
11. Huang, G.B., Zhu, Q., Siew, C.: Extreme learning machine: theory and applications. Neurocomputing **70**, 489–501 (2006)
12. Huang, G.B., Zhou, H., Ding, X., Zhang, R.: Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. **42**(2), 513–529 (2012)
13. Misra, B.B., Dehuri, S.: Functional link artificial neural network for classification task in data mining. J. Comput. Sci. **3**, 948–955 (2007)

14. Goel, A., Saxena, S.C., Bhanot, S.: Modified functional link artificial neural network. Int. J. Electr. Comput. Eng. **1**(1), 22–30 (2006)
15. Anish, C.M., Majhi, B.: Hybrid nonlinear adaptive scheme for stock market prediction using feedback FLANN and factor analysis. J. Korean Stat. Soc. **45**(1), 64–76 (2016)

Application of Computational Intelligence Techniques in the Domain of Net Asset Value Prediction: A Survey

Smita Mohanty and Rajashree Dash

Abstract Extracting meaningful statistics from financial time series data and predicting it effectively and accurately is an emerging and prominent area of research in financial data mining. Out of many financial instruments available till date, an organized and structured investment vehicle is mutual fund. It can be considered as a very common investment strategy in today's financial market as it provides attractive returns with low trading cost and minimum possible risk. The measuring instrument required to price the shares of a fund is known as net asset value (NAV). So NAV prediction can help in strategic decision making by ensuring benefit, capital appreciation, and also in making financial planning. In literature, researchers have suggested many prediction models including linear models, nonlinear models, as well as hybrid models. This study provides a detailed and comparative analysis of various network models used so far by researchers for NAV prediction of top rated firms.

Keywords Data mining · Financial prediction · Financial instrument · Mutual fund · NAV

1 Introduction

Mutual fund is a systematic investment policy where investors invest their savings under a fund. The main idea behind a fund is to lump together the money of small savers. The fund managers manage this money and then invest this accumulated money in numerous capital markets such as stocks, precious metals, property, and other fixed income securities as funds, based on the principle of diversification. They also adjust the fund contained in a specific fund by replacing the poor performing securities with more profitable ones depending on the current market situation. The investors receive specific no. of fund shares in proportion to the amount they invested. They can sell or redeem their fund shares at a later date and pocket in the profit or

S. Mohanty (✉) · R. Dash
Department of Computer Science and Engineering, Siksha 'O' Anusandhan Deemed to Be University, Bhubaneswar, Odisha, India
e-mail: smitamohanty@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_59

loss. So, funds facilitate investment of small amount from investors to make them take benefit from the opportunities of the large, volatile, and fluctuating markets, which are otherwise easily accessible to large investors.

The growth of mutual fund market specifically in India has been tremendous. In this digital age, with the availability of smart gadgets, investors are smartly investing their money in the vast industry as it offers investment of bare minimum and pocket friendly amount. In 1963, this industry has put its very first step in India with the launch of Unit Trust of India (UTI) by the Government of India [1]. Since then with the entry of public sector funds then private sector funds, there is no looking back. So, there is a lot to explore in Indian mutual fund industry. In general, financial market is very chaotic, irregular, and dynamic due to economic, political, and psychological factors. So, predicting time series data like data of mutual fund is usually subject to large errors and a difficult task in hand [2–5]. In literature, many machine learning techniques, integrating computational intelligence systems have been proposed by researchers, which includes artificial neural networks (ANNs), and its variants, support vector machines (SVMs), fuzzy information systems (FIS), and genetic algorithms (GA) for efficient prediction of time series data. The variants of ANN, i.e., multilayer perceptron (MLP), functional link artificial neural network (FLANN), radial basis function neural network (RBFNN), etc., have been widely used prediction tools in recent years. In [6, 7], the prediction capability of SVM with back propagation-based neural network (BPNN) is examined over other models like linear discriminant analysis, quadratic discriminant analysis, and Elman back propagation neural networks. The overall outcome of the experiment proves that SVM is a better alternative for the prediction. A study proposing a hybrid model combining the usefulness of a linear model, i.e., the traditional autoregressive integrated moving average (ARIMA) model and a nonlinear model, SVM is examined in study [8] which provides promising results on real stock price data. An adaptive approach combined with forecasting is examined in another study [9] where multiple inputs and techniques like neural networks and particle swarm optimization (PSO) are used for predicting stock indexes giving high returns. The domain of currency exchange rates time series data prediction is also a promising area of research in finance sector and attracts development of various statistical time series-based models, computational intelligence-based predictor models as well as hybrid models due to its dynamic and nonlinear behavior. In a study [10], Wilcoxon artificial neural network (WANN) and Wilcoxon functional link artificial neural network (WFLANN) are employed to forecast different currency exchange rates for different months ahead. The result of the experiment concludes that both the models perform consistently for different degrees of outliers that exist in the training sample, and the later is preferred over WANN due to its low computational complexity. Keeping in mind the usefulness of computational intelligence (CI) techniques, a comparison has been made in a study [11] between adaptive autoregressive moving average (AARMA) architecture, differential evolution (DE)-based architecture, and a hybrid model combining this two. Out of all these, the AARMA-DE exchange rate predictor model provides superior performance. A knowledge-guided artificial neural network (KGANN) structure is proposed in a paper [12] combining LMS and FLANN model for better prediction of exchange rates. Another study [13] proposes a robust framework integrating an improved shuffled frog leaping (ISFL) algorithm with a CEFLANN model that gives superior prediction performance in comparison with other models that utilizes SFL and PSO algorithm. The ISFL algorithm has also been integrated with a recurrent legendre polynomial neural network (RLPNN) in a study [14] providing higher and efficient predictability for FOREX prediction in comparison with other models used in the study.

In this survey, NAV data prediction of various mutual funds based on the application of soft and computationally intelligent techniques is performed. Rest of this paper is organized as follows: Sect. 1 provides introduction on mutual funds and analysis of evolutionary and CI techniques applied in the field of other time series data prediction like stock and currency exchange rate. Section 2 provides introduction on NAV and motivation behind NAV prediction. Section 3 deals with study of utilization of different neural network-based techniques as well as traditional methods in the domain of NAV prediction. Lastly, the conclusion of this study is drawn in Sect. 4.

2 NAV Prediction

Mutual funds are of different types. Equity funds primarily focus on shares and hence offer greater chances of profit than bonds but also with higher risks. Bonds are considered as fixed income securities. Commodity funds help in investing in the commodity market. Also there are property funds which use major portion of the money invested in buying property, such as land and heritable building rights. Also a distinction can be made between open-end and closed-end funds. Open-ended funds are specialized mutual funds that can be purchased any time and can also be redeemed at any point of time. The freedom in time for investment is a big advantage for such funds, whereas closed-end funds are traded on a stock exchange with a limited term; the shares cannot be returned or increased once the capital investment is done.

Since frequent buy and sell of fund shares start soon after the launch of a particular fund, a measuring instrument is required to price the shares of a fund. This measuring instrument is known as net asset value (NAV). The term “NAV” is most commonly used with respect to mutual funds and is used to determine the value of the assets held. NAV is defined as the net value of a financial instrument or economic entity. It is the value of the fund’s assets minus liabilities or expenses if any and the result is divided by the total number outstanding shares. Thus, NAV represents the price at which investors can buy or redeem the units of the fund on a per-share basis. Most of the popular mutual fund schemes are open ended which facilitates the investors to manage their fund shares independently and decide when to purchase or sell their shares. However, for common man, it is a difficult task to forecast the fluctuation that can happen in highly volatile market like mutual fund. So, NAV prediction is definitely an emerging area of research under finance domain. The next section presents a report on some of these applications.

3 Literature Survey on NAV Prediction

The data-driven and complex data handling features of ANN has made it an immensely popular tool in various fields of research, be it in medicine or in forecasting of financial instruments. Many researchers have utilized ANN models in the field of NAV prediction and have got satisfactory results. In 1996, Chiang et al. used BPNN for NAV forecasting, and the results obtained have been compared with that of the traditional methods like linear regression (LR) and nonlinear regression analysis (NLR) model [15]. The evaluation criteria used were mean absolute percentage error (MAPE), median absolute percent error (MeAPE), geometric mean absolute percent error (GMAPE), maximum absolute percent error (MaAPE), and standard deviation of absolute percent error (STDAPE). The outcome of the study proved that ANNs outperform regression models under less data availability constraints. Another study by Indro et al. in 1999 utilizes a MLP model with a general purpose nonlinear optimizer, GRG2, for analyzing the prediction performance of equity mutual funds [16]. The performance of the predictor model is found to be way better than the performance of the linear models with respect to different types of funds over performance metrics like mean error (ME), mean absolute deviation (MAD), standard deviation of the error (STDDEV), and MAPE. Use of regression analysis method has been done in a paper by E. Priyadarshinil et al. where a comparative analysis is performed for predicting NAV data of Indian mutual fund [17]. The performance analysis in terms of MAPE, root mean squared error (RMSE), mean absolute error (MAE), mean squared error (MSE), and mean percentage error (MPE) has been done using multiple regression analysis (MR) and ANN. Regression analysis (RA) which is a powerful tool for finding out and establishing relationship among independent and dependent variables and ANN due to its flexibility in optimally transforming input have shown interesting results. In 2013, Anish et al. and, in 2015, Ananya et al. applied a forecasting model based on FLANN for NAV forecasting and compared it with MLP model and RBFNN [18, 19]. The property of handling nonlinear data enables FLANN to give better results. A modified version of FLANN, feedback functional link artificial neural network (FFLANN), has been proposed in [20] for NAV prediction and performs better than FLANN and multilayer artificial neural network (MLANN). In [21], time series model like generalized autoregressive conditional heteroskedasticity (GARCH) and ANN has been compared for five months ahead NAV data prediction where GARCH outperforms ANN for all types of funds provided. Use of nature-inspired algorithms like dolphin swarm algorithm (DSA) has been done with extreme learning machine (ELM) in a DSA-ELM model over ELM model for Indian mutual fund data in a paper [22]. Cascaded SVM model with sequential minimal optimization (SMO) technique has been used for prediction of HDFC midcap opportunities (growth) mutual fund over SVM and provides higher flexibility and efficiency [23]. Along with MAE, MAPE, and RMSE, directional accuracy (DA) has been taken as a performance indicator here. A recent work by Anish et al. in 2016 has proposed a hybrid model combining linear models like adaptive moving average (AMA) and AARMA with nonlinear model like feedback radial basis function neural network (FRBFNN) and optimized it with PSO algorithm. For few top rated Indian mutual funds, this model has given superior performance in comparison with other models [24]. In [25], another recent work by Majhi et al. in 2018 proposes an ensemble model consisting of AARMA, AMA, and FLANN based on BPNN and shows very good prediction performance over AARMA, AMA, and FLANN model individually for some Indian NAV data. The brief representation of this survey using CI techniques is presented in Table 1.

4 Conclusion

This survey on NAV prediction throws light on the various CI techniques used so far. Among all these, most of the prior work has been done with the implementation of variations of ANN due to its self-adaptive nature while dealing with complex and nonlinear data. However, some of the recent work deals with hybrid models combining linear and nonlinear models optimized by various learning algorithms. The analysis presented here also provides a brief analysis on various types of performance evaluation metrics that has been used for NAV prediction till date. Among all of them, MAPE and RMSE can be considered as the most popular ones. Research in this area can be extended further with the use of nature-inspired optimization algorithms with ANN models, and also application of fuzzy logic can be exploited to a great extend.

Table 1 CI techniques used in the domain of NAV prediction

Reference no.	Proposed CI techniques	Learning algorithm used	Techniques compared with	Evaluation criteria
[15]	ANN	BP	LR model NLR model	MAPE MeAPE GMAPE MaAPE STDAPE
[16]	MLP model	GRG2 nonlinear optimizer	LR model	ME MAD STDEV MAPE
[17]	ANN	BP	MR model	MAPE RMSE MAE MSE MPE
[18]	FLANN	BP	MLP model	MAPE RMSE
[19]	FLANN	Least mean square (LMS)	MLP RBFNN	MAPE
[20]	FFLANN	BP	MLANN FLANN	MAPE RMSE
[21]	GARCH model	–	ANN	RMSE
[22]	DSA-ELM model	DSA	Standard ELM model	MAPE RMSE
[23]	Cascaded SVM regression model	SMO	–	MAE MAPE RMSE DA
[24]	Hybrid model (AARMA + AMA + FRBF)	PSO	AARMA model, AMA model, FRBF model, ensemble-average model, ensemble-median model, ensemble-trimmed-mean model, ENSM-GA	MAPE RMSE
[25]	Hybrid model (AARMA + AMA + FLANN)	BPNN	AARMA model, AMA model, FLANN	MAPE RMSE

References

1. https://en.wikipedia.org/wiki/Mutual_funds_in_India
2. Tay, F.E., Cao, L.: Application of support vector machines in financial time series forecasting. Omega **29**(4), 309–317 (2001)
3. Zhang, N., Lin, A., Shang, P.: Multidimensional k-nearest neighbor model based on EEMD for financial time series forecasting. Physica A **477**(1), 161–173 (2017)
4. Chen, H., Xiao, K., Sun, J., Wu, S.: A double-layer neural network framework for high-frequency forecasting. ACM Trans. Manag. Inf. Syst. (TMIS) **7**(4), 11:2–11:17 (2017)
5. Zhong, X., Enke, D.: Forecasting daily stock market return using dimensionality reduction. Expert Syst. Appl. **67**(1), 126–139 (2017)
6. Kim, K.: Financial time series forecasting using support vector machines. Neurocomputing **55**(1–2), 307–319 (2003)
7. Huang, W., Nakamori, Y., Wang, S.-Y.: Forecasting stock market movement direction with support vector machine. Comput. Oper. Res. **32**(10), 2513–2522 (2005)
8. Pai, P.F., Lin, C.-S.: A hybrid ARIMA and support vector machines model in stock price forecasting. Omega **33**(6), 497–505 (2005)
9. Chiang, W.C., Enke, D., Wu, T., Wang, R.: An adaptive stock index trading decision support system. Expert Syst. Appl. **59**(1), 195–207 (2016)
10. Majhi, B., Rout, M., Majhi, R., Panda, G., Fleming, P.J.: New robust forecasting models for exchange rates prediction. Expert Syst. Appl. **39**(16), 12658–12670 (2012)
11. Rout, M., Majhi, B., Majhi, R., Panda, G.: Forecasting of currency exchange rates using an adaptive ARMA model with differential evolution based training. J. King Saud Univ. Comput. Inform. Sci. **26**(1), 7–18 (2014)
12. Jena, P.R., Majhi, R., Majhi, B.: Development and performance evaluation of a novel knowledge guided artificial neural network (KGANN) model for exchange rate prediction. J. King Saud Univ.-Comput. Inf. Sci. **27**(4), 450–457 (2015)
13. Dash, R.: An improved shuffled frog leaping algorithm based evolutionary framework for currency exchange rate prediction. Phys. A **486**, 782–796 (2017)
14. Dash, R.: Performance analysis of an evolutionary recurrent legendre polynomial neural network in application to FOREX prediction. J. King Saud Univ. Comput. Inf. Sci. (2017)
15. Chiang, W.C., Urban, T.L., Baldridge, G.W.: A neural network approach to mutual fund net asset value forecasting. Omega Int. J. Mgmt Sci. **24**(2), 205–215 (1996)
16. Indro, D.C., Jiang, C.X., Patuwo, B.E., Zhang, G.P.: Predicting mutual fund performance using artificial neural networks. Omega Int. J. Mgmt. Sci. **27**, 373–380 (1999)
17. Priyadarshinil, E., Chandra Babu, A.: A comparative analysis for forecasting the NAV's of Indian mutual fund using multiple regression analysis and artificial neural networks. Int. J. Trade, Econ. Financ. **3**(5) (2012)
18. Anish, C, M., Majhi, B.: Net asset value prediction using FLANN model. Int. J. Sci. Res. **4**(2), 2222–2227 (2015)
19. Narula, A., Jha, C.B., Panda, G.: Development and performance evaluation of three novel prediction models for mutual fund NAV prediction. Annu. Res. J. Symbiosis Cent. Manag. Stud. **3**, 227–238 (2015)
20. Anish, C. M., Majhi, B.: Prediction of mutual fund net asset value using low complexity feedback neural network. In: International Conference on Current Trends in Advanced Computing (ICCTAC) (2016)
21. Siddiqui, M. U., Abbas, A., AbdurRehman, S. M., Jawed, A., Rafi, M.: Comparison of garch model and artificial neural network for mutual fund's growth prediction. In: International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (2018)
22. Hota, S., Pati, S. P., Mishra, D., Satapathy, P.: Net asset value prediction using extreme learning machine with dolphin swarm algorithm. In: International Conference on Data Science and Business Analytics (2018)
23. Vangara, A., Thouseef, S., Bhat, S.S., Rao, V. V.: Mutual fund NAV prediction using cascaded SVM models. In: International Conference for Convergence in Technology (2018)

24. Anish, C.M., Majhi, B., Majhi, R.: Development and evaluation of novel forecasting adaptive ensemble model. J. Financ. Data Sci. **2**, 188–201 (2016)
25. Majhi, B., Anish, C. M., Majhi, R.: On development of novel hybrid and robust adaptive models for net asset value prediction. J. King Saud University—Computer Inf. Sci. (2018)

An Enhanced 8-Directional Slant Estimation Technique

Mrinaljit Borah, M. K. Ghose, and Debahuti Mishra

Abstract Slant estimation and correction is an integral part of offline handwritten character recognition system. Broadly, it is a part of successful OCR generation. Various techniques such as 4-connectivityand 8-connectivity are being used for the same purpose. However, it is found that majority of these methods are not capable of 100% slant estimation. In this paper, an enhanced slant estimation technique based on 8-directional Freeman chain code is proposed. The proposed technique is tested with Assamese handwritten texts. Results found are found satisfactory and improving.

Keywords Slant estimation · Handwritten character recognition · Slant correction · Word · Image

1 Introduction

The boundary of a digital image can be represented by the movement along the sequence of border pixels by using 8-connectivity or 4-connectivity. Freeman introduced a method, namely Freeman chain code, that describes the contour of a digital image by a connected sequence of straight-line segments with a specific length as well as direction built on 8-connectivity or 4-connectivity of object pixels. The direction of each segment is implied by means of a numbering scheme as given in Fig. 1. In this paper, an enhanced slant [1] estimation technique based on the 8-directional Freeman chain code is proposed.

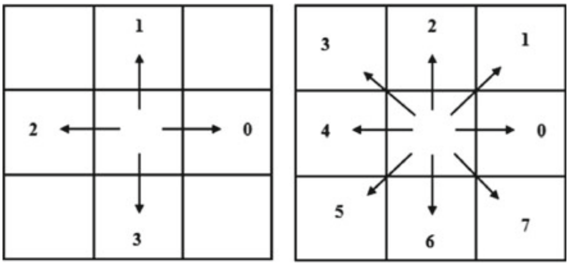
M. Borah (✉)
Department of MCA, Jorhat Engineering College, Garmur, Jorhat, Assam, India
e-mail: mrinalborah23@gmail.com

M. K. Ghose
Department of Computer Science & Engineering, Sikkim Manipal Institute of Technology, Majitar, Rangpo, East Sikkim 737132, India
e-mail: mkghose2000@yahoo.com

D. Mishra
Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, Odisha, India
e-mail: mishradebahuti@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_60

Fig. 1 **a** Freeman chain code with four directions (4-connectivity); **b** Freeman chain code with eight directions (8-connectivity)



2 Related Works

Computational analysis and recognition of handwritten character research is having a rich history. The aim is to develop a successful OCR [2, 2] which is basically a pattern recognition task [3]. Way back, in 1989, Bozinovic and Srihari presented offline cursive script word recognition in their research work [4]. Few more works on cursive handwritten characters can be found in [5–8] discussing skew correction, character segmentation, recognition, etc. Various methods are presented and used by the researchers for slant estimation and correction of handwritten characters such as nonuniform correction [2, 9–14], projection profile-based algorithm [15], word level [16], lexicon directed algorithm [17], core region detection [18], refined chain code [19], etc. An approach for recognizing unconstrained handwritten words is presented by Kimura et al. in [20] using lexicon directed algorithm. Few works are also found for analyzing census data [21, 22].

3 Motivation

The proposed technique is motivated by the work of Ding et al. [19]. They proposed a non-iterative slant estimation method for improving the linearity and the accuracy of the slant estimation of their earlier works. In their method, the boundary of an object was traced with 8-neighburhood and represented as chain code then of 8-directions. Each direction was numbered 0–7 in anticlockwise. The slant estimator of the 8-directional method was given by

$$\theta = \tan^{-1}\left\{\frac{(2n_1 + 2n_2 + n_3) - (n_5 + 2n_6 + 2n_7)}{(n_1 + 2n_2 + 2n_3) + 2n_4 + (2n_5 + 2n_6 + n_7)}\right\}$$

where n_i denotes the number of chain code elements in direction i. Here $(2n_1 + 2n_2 + n_3)$ and $(n_5 + 2n_6 + 2n_7)$ are the sum of the horizontal projection of the element 1, 2, 3 and 5, 6, 7, respectively. The denominator is the summation of vertical estimate of the element 1–7.

3.1 Methodology

The methodology of the abovementioned technique can be represented as follows.

Step 1 Reads an image from file.
Step 2 Converts the image to a grayscale image.
Step 3 Computes threshold of the image using function.
Step 4 Converts to a binary image based on the threshold using function.
Step 5 Performs morphological operation on the binary image using function.
Step 6 Represents contour of the binary image of Step 4 using 8-directional chain code.
Step 7 Counts the number of chain code elements in each direction. Computes the slant angle as

$$\theta = \tan^{-1}\left\{\frac{(2n_1 + 2n_2 + n_3) - (n_5 + 2n_6 + 2n_7)}{(n_1 + 2n_2 + 2n_3) + 2n_4 + (2n_5 + 2n_6 + n_7)}\right\}$$

Step 8 Corrects the slant of the image using shear transformation and returns the slant corrected image.

4 Enhanced 8-Directional Slant Estimation Technique

The proposed enhanced 8-directional slant estimation technique initially represents the contour of a handwritten character using 8-directional Freeman chain code. It computes slant angle of the character by considering the angles that each line segment of chain code makes with the vertical direction. Finally, the slant is corrected using shear transformation.

4.1 Proposed Methodology

The methodology of the given proposed technique can be represented as follows.

Step 1 Reads an image from file.
Step 2 Converts the image to a grayscale image.
Step 3 Computes threshold of the image using function.
Step 4 Converts to a binary image based on the threshold using function.
Step 5 Performs morphological operation on the binary image using function.
Step 6 Represents contour of the binary image of Step 4 using 8-directional Freeman chain code.

Step 7 Computes the angle that each line segment of chain code makes with the vertical direction. Take the average of the angles as estimated slant angle.

Step 8 Corrects the slant of the image using shear transformation and returns the slant corrected image.

4.1.1 Illustration

Let us consider the input image as given in Fig. 2.

This input image is transformed to grayscale image. Figure 3 shows the grayscale image.

Before converting to binary image, a global threshold value of the grayscale image is calculated. Based on the threshold, the grayscale image is converted to binary image. In this conversion, all pixels in the grayscale image having luminance greater than threshold are replaced with the value 1 (white) and all other pixels are replaced with the value 0 (black). Figure 4 shows the binary image.

Fig. 2 Input image

Fig. 3 Grayscale image

Fig. 4 Binary image

For ease of tracing the boundary, the binary image is thinned and removed the isolated pixels from the image. The resulting Fig. 5 is given.

The boundary of the filtered image is traced with 8-neighborhood. The boundary is represented as 8-directional Freeman chain code. Each direction is numbered 0–7 in clockwise as given in Fig. 6.

The tracing is started from the leftmost object pixel (pixel value 1) and is considered as the current pixel. Now the 8-neighborhood of the current pixel is checked to find out the next object pixel. The tracing is done in clockwise direction. The object pixel that is traced first is considered as the new current pixel, and its direction is recorded with the code given in Fig. 6. The angle that the line segment between the pixels old current and new current makes with the vertical direction is calculated and

Fig. 5 Filtered image

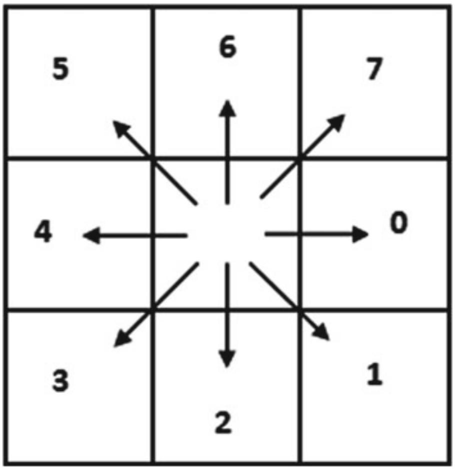


Fig. 6 Chain code in 8-direction

Fig. 7 Slant corrected image

recorded separately. In this way, the entire border pixels of the image are traced and each time the direction code as well as the angle is recorded. The average of the angles is considered as estimated slant angle and is corrected by shear transformation. The slant corrected image is given in Fig. 7.

4.2 Algorithm with Sub-functions

The algorithm of the enhanced 8-directional slant estimation technique is designed as a set of three functions. The first function is the overall design of the algorithm. It is used to preprocess the input image before slant estimation and as well as to call the second and third function. The second function is used to detect the slant of the input image, and the third function is used to correct the slant of the image.

Algorithm slant_en (img_path)
/* It takes the path of the image as input and read the image from the file given by the image path. The input image ‘I’ is a Red Green Blue (RGB) image. ‘J’ is the grayscale image of ‘I’, and ‘level’ is the global threshold based on which binary image ‘BW’ is formed. Here, ‘z’ is the filtered image and ‘conn’ is the connectivity. The ‘z’ and ‘conn’ are used as the input arguments for the get_slant(). */

Input Path of the image as img_path
Output Slant corrected image

1. **begin**
2. I ← imread (img_path)
3. J ← rgb2gray (I)
4. level ← graythresh (J)
5. BW ← im2bw (J, level)
6. a ← ~ BW
7. z ← bwmorph (a, ‘thin’, 5)
8. z ← bwmorph (z, ‘clean’, 5)
9. conn ← 8
10. slant_angle ← get_slant (z, conn)
11. slant_correct (z, slant_angle)
12. **end**

Function get_slant (z, conn) /* This function is the most important part of entire algorithm. It takes the filtered image ‘z’ and connectivity ‘conn’ as input arguments and computes the slant angle of the image. It traces the boundary of the image ‘z’ with 8-neighborhood and represents as 8-directional Freeman chain code. It computes the angle that each line segment of the chain codes makes with the vertical direction and takes the average of the angles as estimated slant angle. */

Input Filtered image z, connectivity conn (= 8)
Output Slant angle

1. **begin**
2. table ← [0 -1; -1 -1; -1 0; -1 1; 0 1; 1 1; 1 0; 1 -1]
3. bwn ← zeros (size(z) + [2 2])
4. bwn (2 length (bwn(, 1)) – 1, 2 length (bwn(1,)) - 1) ← z
5. z ← bwn
6. offset ← conn/2
7. [l, num] ← bwlabel (z, conn)
8. all_code ← [ ]
9. all_angle ← [ ]
10. tot_angle ← 0
11. n ← 1
12. **for**i = 1 to num

    12.1. code ← [ ]
    12.2. angle ← 0
    12.3. [r, c] ← find (bwlabel (z) == i, 1, 'first')
    12.4. start ← [r, c]
    12.5. current ← start
    12.6. count ← 1
    12.7. index ← 2
    12.8. Repeat **while** 1 == 1

        12.8.1. xy ← current + table (index + 1,)
        12.8.2. Repeat **while** z (xy(1), xy(2)) == 0

            12.8.2.1. index ← mod (index + 1, conn)
            12.8.2.2. xy ← current + table (index + 1,)

            [End of while loop]

        12.8.3. code (count) ← mod (index + offset, conn);
        12.8.4. all_code(n) ← code (count)
        12.8.5. previous ← current
        12.8.6. x1 ← previous(1, 1)
        12.8.7. y1 ← previous(1, 2)
        12.8.8. current ← current + table (index + 1,)
        12.8.9. x2 ← current(1, 1)
        12.8.10. y2 ← current(1, 2)
        12.8.11. x3 ← current(1, 1)
        12.8.12. y3 ← previous(1, 2)
        12.8.13. d1 ← abs (sqrt((x2 - x1)**2 + (y2 - y1)**2))
        12.8.14. d2 ← abs (sqrt((x3 - x2)**2 + (y3 - y2)**2))
        12.8.15. m ← (y2 - y1) / (x2 - x1)
        12.8.16. **if** m < 0 **then**

            12.8.16.1. k ← 1

```
                    else
            12.8.16.2.      k  ←  -1
                    [ End of if structure]
        12.8.17.        slant  ←  k * asin (d2/d1)
        12.8.18.        angle  ←  angle + slant
        12.8.19.        index  ←  mod (code(count) + 1, conn)
        12.8.20.        n  ←  n + 1
        12.8.21.        if current == start
            12.8.21.1.      all_angle (i)  ←  angle
            12.8.21.2.      Go to Step 12
                    [ End of if structure]
        12.8.22.        count  ←  count + 1
                [ End of while loop]
        [ End of for loop]
13.   fori = 1 to num
    13.1. tot_angle  ←  tot_angle + all_angle (i)
    [ End of for loop]
14. number  ←  length (all_code)
15. avg_angle  ←  tot_angle / number
16. end
```

Function slant_correct (z, slant_angle) /* This function takes the image 'z' and the slant angle as input parameters and outputs a slant corrected image. */

Input Filtered image 'z', slant slant_angle
Output Slant corrected image

1. **begin**
2. slope ← tan (slant_angle)
3. t ← maketform ('affine', [1 0 0; slope 1 0; 0 0 1])
4. img ← imtransform (z, t)
5. SE ← strel ('line', 4, 0)
6. L ← imdilate (img, SE)
7. imshow (L)
8. **end**

5 Implementation

The proposed algorithm is implemented using MATLAB on windows platform. The test database has been prepared during sample collection phase. The various functions developed for this implementation are given below:

1. slant_en (img_path)
2. get_slant (z,conn)
3. slant_correct (z, slant_angle).

Fig. 8 Original image

In the implementation, the following inbuilt functions are also used.

1. imread (img_path)
2. rgb2gray (I)
3. graythresh (J)
4. im2bw (J)
5. bwmorph (a, 'thin', 5) &bwmorph (z, 'clean', 5)
6. bwlabel (z, conn)
7. find (bwlabel (z) == i, 1, 'first')
8. maketform ('affine', [1 0 0; slope 1 0; 0 0 1])
9. imtransform (z, t)
10. strel ('line', 4, 0)
11. imdilate (img, SE)
12. imshow (L).

To execute the application, user can go to the MATLAB command window and type the command $f_x \gg$ **slant_en ('C\Users\Mrinal\Desktop\scan\scan1.jpg')**

where 'C\Users\Mrinal\Desktop\scan\scan1.jpg' is the complete path of the file that contains the input image scan1.jpg. The results are given in Figs. 8, 9, 10 and 11.

6 Result and Discussion

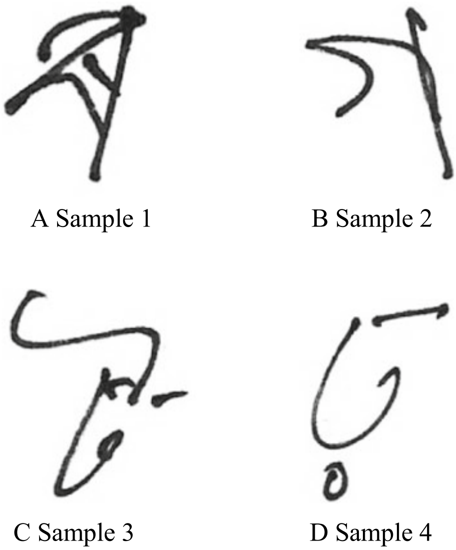
The 8-directional slant estimation technique proposed by Ding et al. [19] has been implemented in MATLAB. This technique is used for performance comparison with the proposed slant estimation technique. It is applied to the dataset of slanted handwritten characters of different individuals that had been prepared during sample collection phase. The both techniques are applied on four sample image data of slanted characters and observed the result accordingly.

Fig. 9 Binary image

Fig. 10 Filtered image

Fig. 11 Slant corrected image

Fig. 12 Sample images



A Sample 1
B Sample 2
C Sample 3
D Sample 4

6.1 Input Data

See Fig. 12.

6.2 Results

The 8-directional slant estimation technique proposed by Ding et al. [19] gives the following results.

1. Results of the existing technique (Ding et al.)

Result for Sample 1
See Fig. 13.

Actual slant = 12.50º
Estimated slant = 7.90º
Percentage of accuracy in slant estimation = 63.20%.

Result for Sample 2
See Fig. 14.

Actual slant = 15º
Estimated slant = 7.18 º
Percentage of accuracy in slant estimation = 47.86%.

Fig. 13 Slant corrected image (by Ding et al.)

Fig. 14 Slant corrected image (by Ding et al.)

Result for Sample 3

See Fig. 15.

Actual slant = 23.50º
Estimated slant = 20.38º
Percentage of accuracy in slant estimation = 86.72%.

Result for Sample 4

See Fig. 16.

Actual slant = 24º
Estimated slant = 15.71 º
Percentage of accuracy in slant estimation = 65.45%.

2. Results of the proposed technique

Fig. 15 Slant corrected image (by Ding et al.)

Fig. 16 Slant corrected image (by Ding et al.)

The proposed slant estimation technique gives the following results.

Result for Sample 1
See Fig. 17.

Actual slant = 12.50°
Estimated slant = 10.10 °
Percentage of accuracy in slant estimation = 80.80%.

Result for Sample 2
See Fig. 18.

Actual slant = 15°
Estimated slant = 11.25 °
Percentage of accuracy in slant estimation = 75%.

Result for Sample 3
See Fig. 19.

Fig. 17 Slant corrected image (by proposed technique)

Fig. 18 Slant corrected image (by proposed technique)

Actual s = 23.5º
Estimated slant = 19.77º
Percentage of accuracy in slant estimation = 812%.

Result for Sample 4 See Fig. 20.

Fig. 19 Slant corrected image (by proposed technique)

Fig. 20 Slant corrected image (by proposed technique)

Actual slant = 24º
Estimated slant = 19.48º
Percentage of accuracy in slant estimation = 81.16%.

In the above observations, the actual slants of the four input images are calculated. The results obtained from the 8-directional slant estimation technique of Ding et al. [19] and the proposed slant estimation technique are summarized in Table 5.1.

7 Conclusion

In handwritten word recognition, slant estimation and correction is a widely used preprocessing technique to improve segmentation performance, which significantly affects recognition accuracy. In this paper, an enhanced slant estimation technique based on 8-directional Freeman chain code is proposed. The algorithm is implemented in MATLAB. The proposed technique is applied on the dataset and found desired results.

References

1. Slant: http://www.thefreedictionary.com/slant
2. Kavallieratou, E., Fakotakis, N., Kokkinakis, G.: Slant estimation algorithm for OCR systems. Pattern Recogn. **34**, 2515–2522 (2001)

3. Devijver, P.A., Kittler, J.: Pattern Recognition: A Statistical Approach. Prentice Hall, London (1982)
4. Bozinovic, R.M., Srihari, S.: Off-line cursive script word recognition. IEEE Trans. Pattern Anal. Mach. Intell. **11**(1), 68–83 (1989)
5. Yan, H.: Skew correction of document images using interline cross-correlation. Comput. Vis. Graph. Image Process. **55**, 538–543 (1993)
6. Lu, Y., Shridhar, M.: Character segmentation in handwritten words. Pattern Recogn. **29**(1), 77–96 (1996)
7. Guillevic, D., Suen, C.Y.: Cursive script recognition: a sentence level recognition scheme. In: Proceedings of 4th IWFHR, pp. 216–223, Dec 1994
8. Negi, A., Swaroop, K.S., Agarwal, A.: A correspondence based approach to segmentation of cursive words. In: Proceedings of 3th ICDAR, vol. II, pp. 1034–1037, Aug 1995
9. Bertolami, R., Uchida, S., Zimmermann, M., Bunke, H.: Non-uniform slant correction for handwritten text line recognition, In: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). IEEE, 0-7695-2822-8/07 $25.00 © 2007
10. Taira, E., Uchida, S., Sakoe, H.: Nonuniform slant correction for handwritten word recognition. IEICE Trans. Inf. Syst. **E87-D**(5), 1247–1253 (2004)
11. Vinciarelli, A., Luettin, J.: New Techniques for both Slant and Slope Angle Estimation (2001)
12. Ding, Y., Kimura, F., Miyake, Y., Shridhar, M.: Accuracy Improvement of Slant Estimation for Handwritten Words
13. Ding, Y., Kimura, F., Miyake, Y., Shridhar, M.: Evaluation and improvement of slant estimation for handwritten words. In: Proceedings of 5th ICDAR, pp. 753–756, Sept 1999
14. Britto, A.D.S. Jr., Sabourin, R., Lethelier, E., Bortolozzi, F., Suen, C.Y.: Improvement in handwritten numeral string recognition by slant normalization and contextual information
15. Pastor, M., Toselli, A., Vidal, E.: Projection profile based algorithm for slant removal. In: Proceedings of the International Conference on Image Analysis and Recognition, pp. 183–190 (2004)
16. Kavallieratou, E., Fakotakis, N., Kokkinakis, G.: New Algorithms for Skewing Correction and Slant Removal on Word-Level. IEEE, 0–7803-5682-9/99/$10.0001999
17. Kimura, F., Shridhar, M., Chen, Z.: Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words. In: Proceedings of the 2nd ICDAR, pp. 18–22 (1993). X
18. Rehman, A., Mohammad, D., Sulong, G., Saba, T.: Simple and effective techniques for core-region detection and slant correction in offline script recognition. In: 2009 IEEE International Conference on Signal and Image Processing Applications
19. Ding, Y., Kimura, F., Miyake, Y., Shridhar, M.: Slant estimation for handwritten words refined chain code
20. Kimura, F., Shridhar, M., Chen, Z.: Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words. In: International Conference on Document Analysis and Recognition, vol. 93, pp. 18–22 (1993)
21. Simoncini, L., Kovacs-V, Z.M.: A system for Reading USA census "90 hand-written fields". In: Proceedings of 3th ICDAR, vol. II, pp. 86–91, Aug 1995
22. Simoncini, L., Kovacs-V, Z.M.: A system for Reading USA census' 90 hand-written fields. In: Proceedings of 3rd ICDAR, vol. II, pp. 86–91, Aug 1995

Astute Security and Alert Proposition for Women Security

Biswaranjan Swain, Debashish Naik, Jayshree Halder, and Satyanarayan Bhuyan

Abstract In the current global scenario, one question that arises in every woman's mind is their safety and security. The crime rates against women are increasing at an alarming rate. With the advancement in technology, it is now possible to provide security and self-defense mechanism to women in more than one way. When safety and security are concerned, a smartphone can become a powerful tool to prevent violence against women. Having an app with security features for women, the smartphone tries to help women by sending emergency alerts to chosen people and also lets the police know about victim position and hope nothing goes wrong. Keeping this in mind, an android app named "DEV Ashish" has been developed in Java Development Kit using Android Studio, which is dedicated to providing respite to the relatives in trouble and anxiety. By clicking on a button that is provided on the app, an alert message is sent to the user's emergency contacts. The application shares the user's location with the registered emergency contacts in the form of a message. The application has key features like "alarming as police siren," "fake call," "panic SMS to emergency contacts," "anonymous complaints to five different women safety departments," "offline location sharing to the nearby police station along with IMEI number," etc.

Keywords Astute women security · Smartphone · Android app · Alert message

B. Swain (✉) · J. Halder · S. Bhuyan
Department of Electronics and Communication Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: biswaranjanswain@soa.ac.in

D. Naik
Department of Computer Science Engineering, Siksha 'O' Anusandhan University (Deemed to be University), Bhubaneswar, Odisha 751030, India

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_61

1 Introduction

Women are given special honor and respect, according to the tradition of India. Special care is taken of the safety and admiration of women in India. In the twenty-first century, women are working on the shoulders with men in every field, whether it is bank, sports, school, politics, police, defense, own business, or the desire to fly in the sky is. According to Swami Vivekananda, "the best thermometer to the progress of a nation is its treatment of its women." Violence against women has taken many forms in this society and nowadays increase in sexual violence, as well as a fundamental violation on women's rights. Violence in social, economic, developmental, legal, educational, human rights, and health (physical and mental) has shamed humanity as whole, and now being a woman is considered as a curse for them [1].

According to WHO, in all over the world, 40% of women are facing numerous immoral bodily nuisances everyday even in public spaces such as foot paths, bus stands, railway stations, etc. [2]. The United Nations General Assembly describes "violence against women" as performance of gender-based ferocity that consequences in physical, mental, or sexual damage or grief to women, including intimidations of such acts, pressure, or arbitrary lack of freedom, whether occurring in public or in private lifespan. [3]. On November 27, 2019, the heart wrenching gang rape of the 26-year-old veterinary doctor which was followed by an inhuman act of burning the victim alive in Shamshabad, near Hyderabad, shook every single human heart [4–6].

Hence, there should be a system which can at least can help a woman to tell about their whereabouts and can aware the police and relatives if they are in danger. This is a simple application which can be installed in smartphone, which almost every woman carries all the time, by a developed application to help women in such emergency situation [7–9]. This paper describes a very reliable security application named "DEV Ashish" which will be there as a companion with women at emergency situation. The app got activated simply by shaking your smartphone or clicking volume button rapidly or simply clicking the app icon. By touching a simple button which is provided on the app will send a message to the user's emergency contacts. User's location will be instantly shared to the registered emergency contacts and police in the form of message. The application has key features like "alarming as police siren," "fake call," "panic SMS to emergency contacts," "anonymous complaints to five different women safety departments," "offline location sharing to nearby police station along with IMEI number," etc.

2 Features of "DEV Ashish" App

The key features of the app "DEV Ashish" are listed below, which provides an overview of the security provided by the application system.

The app can be used for various purposes:

1. **SOS**: On clicking the SOS button, an SMS "I'm in Danger" with your current location is obtained by goggle play service online and offline tool to three emergency contact saved while setting up the app first time and an SMS "I'm in Danger" with your current location phone IMEI number SIM number to police and call to 100.
2. **Profile**: On click of profile button, it helps to reset the emergency contact and later can be used as panic messenger, i.e., send an SMS "I'm not feeling right" with your current location obtained by Goggle play service online and offline tool to three emergency contacts.
3. **Fake Call**: Fake call button helps on situation when someone tries to start an unwanted conversation the app can help you to generate a fake call to talk.
4. **Alarm**: Alarm button can be clicked when feel like being followed alarm can help you to play police siren on full volume to scare of the followers.
5. **Complaints**: This button is clicked when women feel a need to file a complaint via email anonymously to concerned department for different problem faced by women's like domestic violence, eve teasing, corruption, theft attempt, threat.

3 Experimental Setup

The experiment has been done using Android SDK with 1 GB space, available disk space (minimum) of 2 GB, and with 8 GB RAM of Intel (R) Core (TM) i5 4210U CPU at 1.80 GHz processor. The screen resolution has been maintained up to 1280 $\times$ 800 (minimum) and Windows 7 operating system of either (32-bit or 64-bit) has been used. The different software that has been used in this experiment are Android Studio 3.5.2 and Java Development Kit 8 or higher.

4 UML Diagrams

The use case diagram provided here is a graphical representation of the interactions among the elements of the system (Figs. 1, 2 and 3).

5 Results and Discussion

In this section, few output pages from the developed "DEV Ashish" app have been shown along with its main features.

1. First page is a SPLASH screen which displays the logo of the app, which appears for 5 s before loading the main user interaction screen. The same logo is the icon on mobile screen which is shown in Fig. 4.

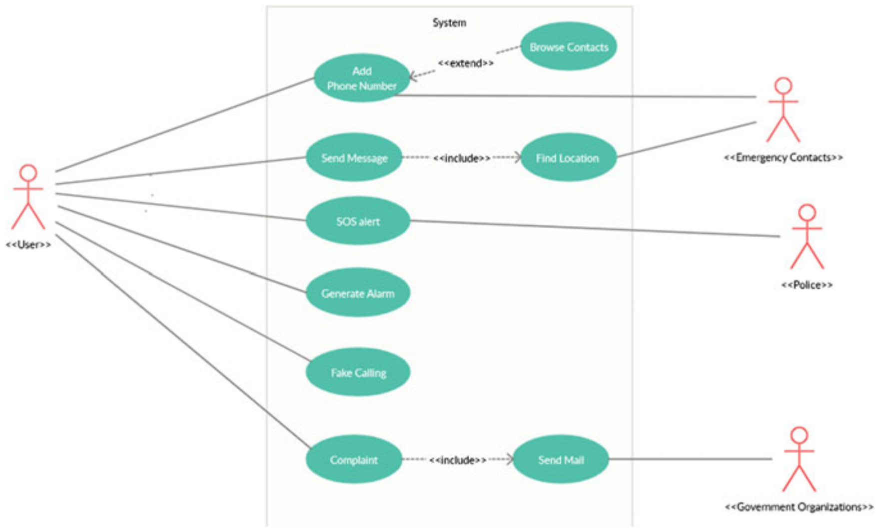


Fig. 1 Use case diagram of the system

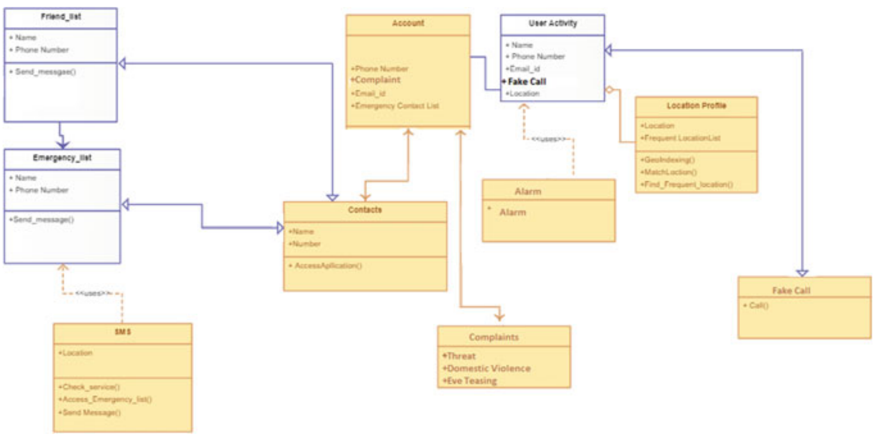


Fig. 2 Class diagram of the system

2. Fake call button helps on situation when someone tries to start an unwanted conversation the app can help you to generate a fake call to talk (Figs. 5, 6, 7, 8 and 9).

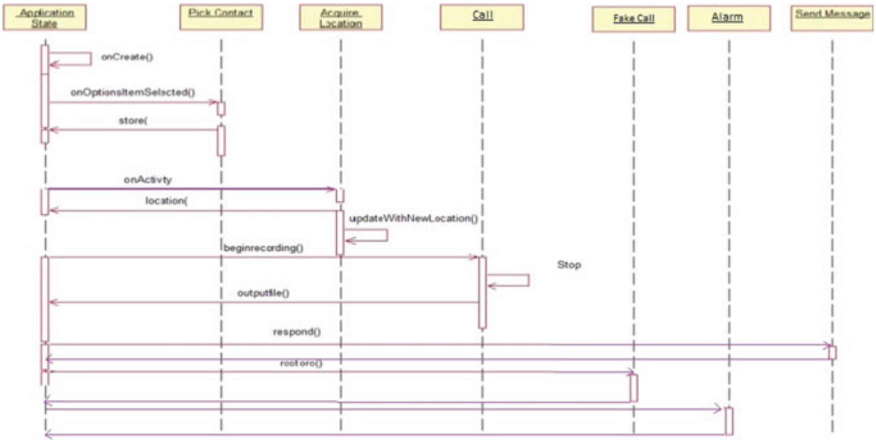
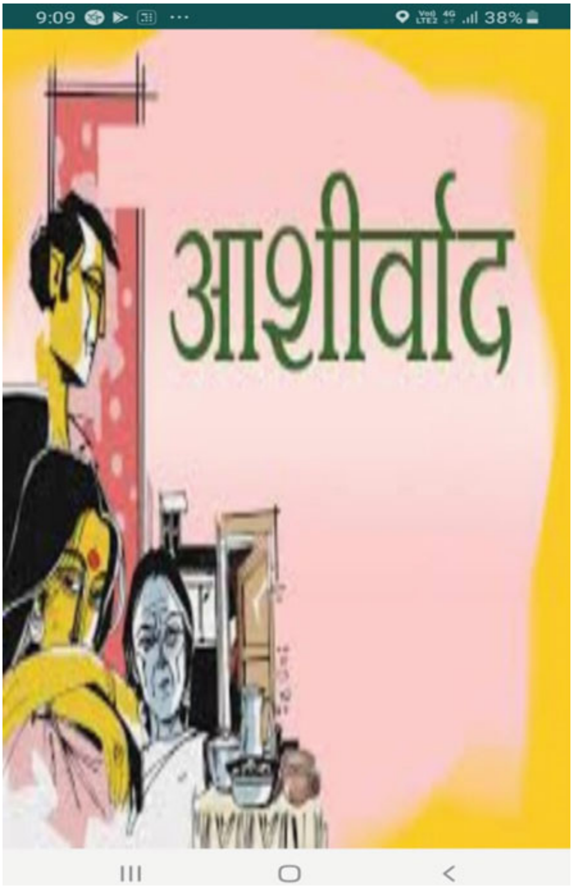


Fig. 3 Interaction diagram of the system

Fig. 4 Logo of the app



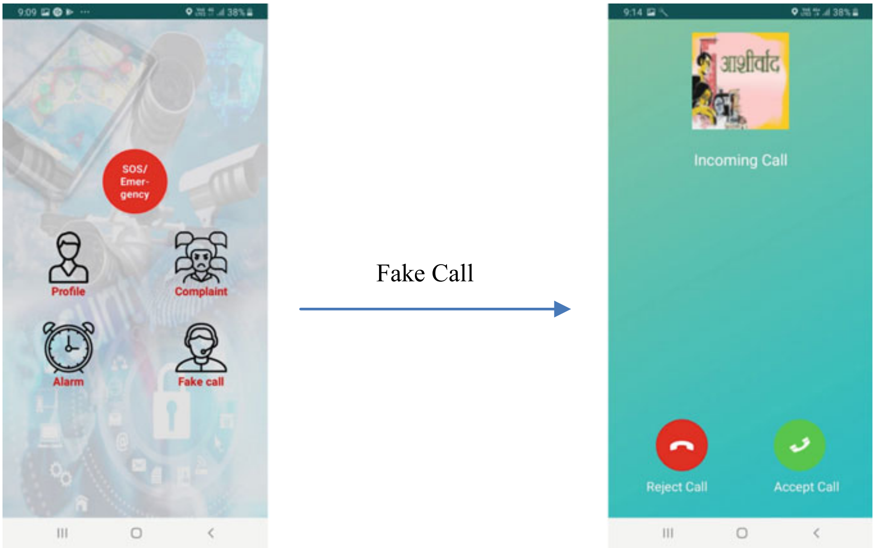


Fig. 5 Pressing fake call

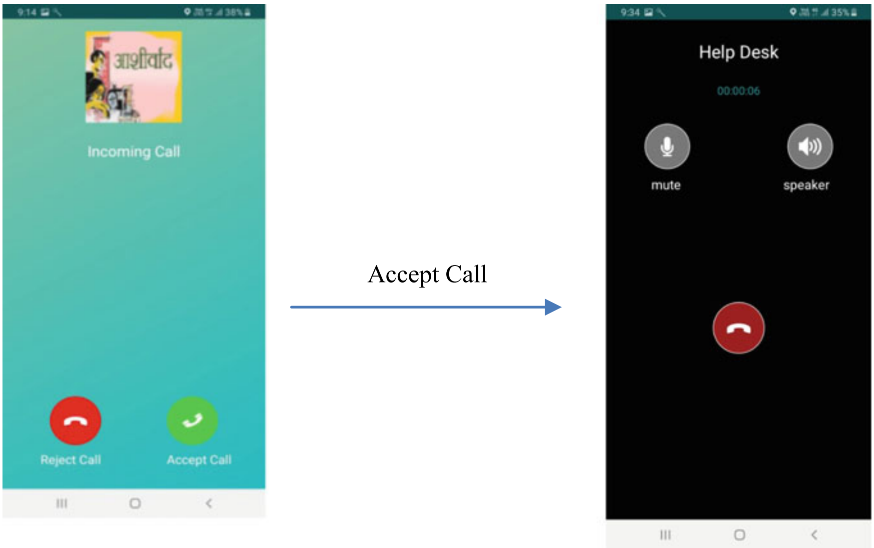


Fig. 6 Pressing accept call

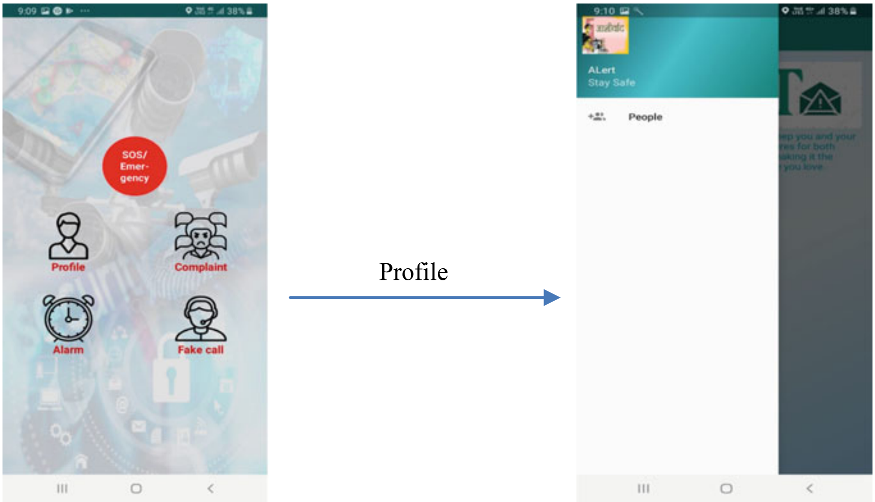


Fig. 7 Pressing profile button

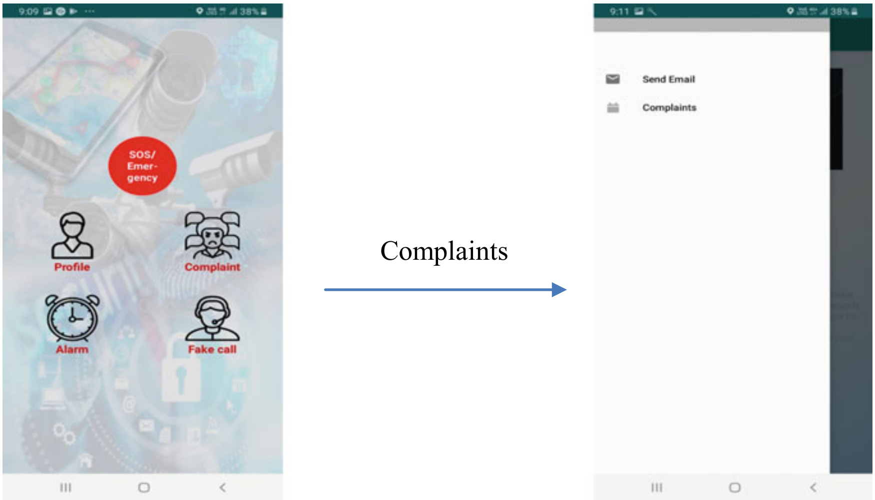


Fig. 8 Pressing complaint button

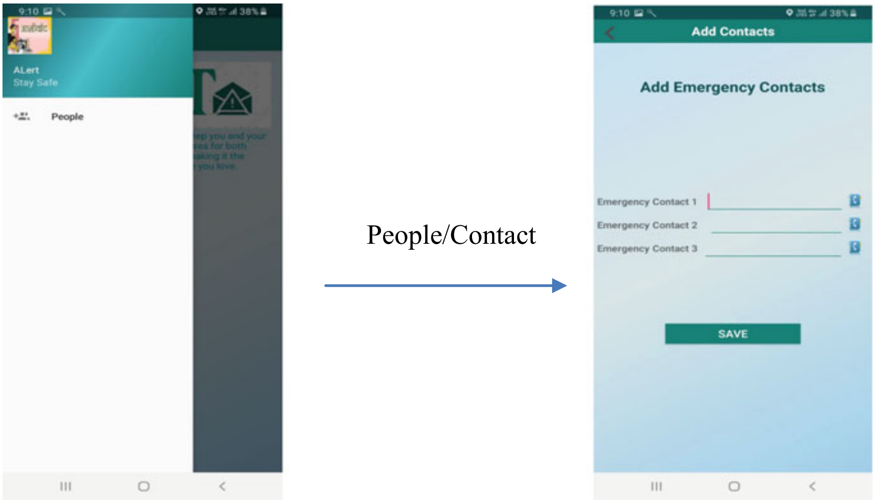


Fig. 9 Pressing people/contact button

6 Conclusion

"DEV Ashish" app tries to provide a safety companion to all the women in this society and allows them to move around with a sense of someone being with them 24 × 7. Women can send their location to their family members and police station by clicking SOS options. It is an application which tries to minimize risk to extent for women. Until and unless there is no strict law in this country, apps like these are of no use until it can let women shoot the abuser with the help of smartphone. With further research and innovation, this app in future can be extended to safety and wellbeing of women's in our society such as using emotion detector technology sensing the emotions of the user and auto-activate the corresponding required action, voice sensing of surrounding and activated through key abusive words, external hardware connection to respond to attack and threats, etc.

References

1. Poddar, T., Ritesh, C., Bharath, N.: Using wearable technology to answer women's safety. Int. J. Sci. Technol. Manage. **04**, 176–180 (2015)
2. Saranya, N., Karthik, K.: Women safety application using android mobile. Int. J. Eng. Sci. Comput. **03**, 1317–1319 (2015)
3. Pasha, S., Kavana, J., Mangala, G.K.R., Nischitha K., Surendra, B.K., Rakshitha, M.S.:B secure for women: an android application. Int. J. Innov. Res. Comput. Commun. Eng. **04**, 8073–8080
4. Thota, B., Kumar, U.K.P.: Sauver: an android mobile for women safety. Int. J. Technol. Enhance. Emerging Eng. Res. **03**, 122–126 (2015)

5. Pawar, V., Wankhade, N.R., Nikam, D., Jadhav, K., Pathak, N.: SCIWARS android app for women safety. Int. J. Eng. Res. Appl. **04**, 823–882 (2014)
6. Mandapati, S., Pamidi, S., Ambati, S.: A mobile based women safety application (I safe apps). IOSR J. Comput. Eng. **17**, 29–34 (2015)
7. Uma, D., Vishakha, V., Ravina, R., Rinku, B.: An android application for women safety based on voice recognition. Int. J. Comput. Sci. Mobile Comput. **04**, 216–220 (2015)
8. Paradkar, A., Sharma, D.: All in one intelligent safety system for women security. Int. J. Computer Appl. **130**, 1123–1126 (2015)
9. Sharma, K., More, A.: Advance woman security system based on android. In. J. Innov. Res. Sci. Technol. **2**, 3901 (2015)

Morph Analyzer of Verbs in Nepali Language

Pema Yangzom Bhutia, Samarjeet Borah, and Debahuti Mishra

Abstract Morph analyzer is a tool which deals with forming of words in any natural language. It also highlights the relationship of any word to other words in a language. Relationship among words in inflection and derivation is common in Nepali. In this paper, a study on various works on morph analysis from the literature is presented. The main objective is to focus on the variation aspect of verb morphology and the analysis of verb stems from a given sentence/speech.

Keywords Nepali verbs · Morph analyzer (MA) · Nepali language · Natural language processing (NLP) · Morphology · Inflection

1 Introduction

Natural language processing (NLP) is a part of artificial intelligence (AI) where we can train the system the human languages to interact with the machine. We train the computer by how to program or process the large amount of natural language data to understand the human language for interactions between human and computer system. NLP is one of the mostly commonly research tasks to show the syntactic and semantic analysis of the languages which includes part of speech tagging, morphological analysis, machine translation, sentiment analysis, word segmentation, parsing, stemming, etc.

P. Y. Bhutia (✉) · S. Borah
Department of Computer Applications, Sikkim Manipal Institute of Technology, Sikkim Manipal University, Majitar, Rangpo, East Sikkim 737132, India
e-mail: yangzom0895@gmail.com

S. Borah
e-mail: samarjeetborah@gmail.com

D. Mishra
Department of Computer Science & Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: mishradebahuti@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_62

Nepali is an Indo-Aryan language which belongs to the Indo-European family of languages. It is approximately spoken by 17 million people and is written using the Devanagari script. It follows a Subject + Object + Verb pattern in sentences. That means the verb comes at the end of the sentences. The verbs in Nepali inflect for the tense, aspect, mood, honorific, gender, person, and number. This language differs normally when spoken from the way it is written. The Nepali language writing system is Devanagari which is also common to other languages like Hindi and Marathi with some differences.

Morphological variations in most languages can be attributed to the inflection and derivation methods for word formation. By inflection, we understand that the word is formed by the combination of morphological variants of a lexeme that will show same syntactic category which will also reflect the change based on such relations. Inflections are achieved mainly through attaching prefixes and suffixes to the roots. Derivational morphology is those where words can be generated with different meanings and different syntactic categories. In this paper, our focus remains on the inflectional morphology and suffixation for verbs.

The main objective of this research work is to develop a morphological analyzer for Nepali verbs which is based on rule-based systems. A rule-based system is a way of encoding a human expert's knowledge in a fairly narrow area into an automated system. A rule-based system can be simply created by using a set of assertions and a set of rules that specify how to act on the assertion set. Rules are expressed as a set of if-then statements (called IF-THEN rules or production rules) [1].

1.1 Verbs in Nepali

A verb is a word that indicates an action, event, or state. Verbs in Nepali are called 'क्रियापद' (*kriyapad*); it defines the work to do or to be done. The Nepali verbs inflect to show persons (first, second, third), tense (past, present, future) for persons, gender of subject in third person plural (masculine, feminine), singular and plural numbers, grades of honorifics in second and third persons (Table 1).

Table 1 Grammatical categories of verb inflections in Nepali

Gender	Masculine	Feminine		
Number	Singular	Plural		
Tense	Past	Present	Future	
Mood	Imperative	Optative	Potential	
Person	First	Second	Third	
Honorifics	Low/Casual	Medial/Familiar	High/Respectful	
Aspect	Perfect	Imperfect	Habitual	Inferential

2 Literature Survey

Some of the important works based on language and/or computational approaches available are reviewed for the current research work. There is not enough computational work on Nepali morphology. Few of the works from linguistic as well as computational domain on Nepali language are included in this paper.

Work on linguistic point of view is available in the literature with a morph analyzer [2]. Subsequently, the developed morph analyzer is capable of POS tagging, chunking, syntactic parsing, word sense, etc. A work on morphological analysis of Nepali verbs is found at [3]. The work is having modules to generate different verb forms. Segmentation of the verb word is also presented. A compiled list of roots and suffixes is used to apply on rule-based analysis. Morph analyzers for nouns and non-declinable adjectives of Nepali language using finite-state grammar are also available [4, 5]. A hidden Markov model-based POS tagging approach is available in [6]. Another work on POS tagging of Nepali language is available in [7]. Balaram Prasain's work presents a morphological analyzer for developing a model based on computational aspects of morphology in basic Nepali verbs [8].

Few more works from the NLP domain of other languages are included in the paper to portrait a picture on development scenario. A work on Bengali verb subcategorization frame acquisition can be found in [9]. Sub-categorization of verb frames is performed. The concept is based on that the verbs in a sentence take different types of arguments that are associated with each sentence during comparisons with other part of speech (POS) categories. A study on morphological analysis of Indian and European languages can be found in [10]. It highlights on the design and development of a morphological analysis (MA) of morphological rules from the root words. Morph analyser for Malayalam verbs is also under development [11].

During the conduction of literature review, few important points are observed. The key issue is a smaller number of computational works are there on Nepali morphology. A need and maintenance of corpus and dictionary before the actual development of morph analyser. These are based on semantic features, therefore cannot tag new words if not found in the dictionary.

3 The Proposed Work

Nepali language is rich in morphology which results in a lean work from computational point of view. Moreover, morphology of Nepali verbs has not yet been fully analyzed from computational perspective. In this regard, the problem for this research work can be defined as to design and develop a suffixal morph analyzer for Nepali verbs.

3.1 Objectives

The main objective of this research work is to develop a suffixal morph analyzer for Nepali verb based on finite state grammar approach. The developed morph analyzer (MA) correctly tags the tokens based on their respective suffixes. The objective of this study is to analyze the morphology in Nepali verbs from computation perspective. The goal of my research work is to detect the verb from the given sentence, assign its respective tag and show its respective suffixes.

3.2 Methodology

For developing morph analyzer, a transliteration tool is developed for converting the text from Devanagari to Roman. Linguistic approach is followed that is based on the grammar of language; it also includes the pattern matching and finite state grammar to show the sequencing of grammars required for the inflection of words. For transliterating each character from Devanagari to Roman, the transliteration scheme of Linguistic Data Consortium for Indian Language (LDC-IL) has been used.

To get successful tags for the verbs, rules of finite state grammar are followed. The available tags for its respective tokens are:

VF Verb Finite

VVMX1	First Person Singular Verb
VVMX2	First Person Plural Verb
VVTN1	Second Person Non-Honorific Singular Verb
VVTX2	Second Person Plural (or Mid-Honorific Singular) Verb
VVYN1	Third Person Non-Honorific Singular Verb
VVYX2	Third Person Plural (or Mid-Honorific Singular) Verb
VVTN1F	Feminine Second Person Non-Honorific Singular Verb
VVTM1F	Feminine Second Person Mid-Honorific Singular Verb
VOMX1	First Person Singular Optative Verb
VOMX2	First Person Plural Optative Verb
VOTN1	Second Person Non-Honorific Singular Optative Verb
VOTX2	Second Person Plural (or Mid-Honorific Singular) Singular Optative Verb
VOYN1	Third Person Bon-Honorific Singular Optative Verb
VOYX2	Third Person Plural (or Mid-Honorific Singular) Optative Verb

VI/VBI Verb Infinitive
VBN Prospective Verb
VBKO Aspect Verb

VDM	Masculine d-participle Verb
VDF	Feminine f-participle Verb

VDO	Other—agreement d-participle Verb
VDX	Unmarked d-participle Verb

VBO Others

VE	e(ko)-participle Verb
VCN	Command-form Verb, Non-Honorific
VCM	Command-form Verb, Mid-Honorific
VCH	Command-form Verb, High-Honorific
VS	Subjunctive/Conditional e-form Verb
VQ	Sequential Participle-converb
VR	i-form Verb

3.3 Design of the Proposed Morph Analyser

See Fig. 1.

4 Implementation Details

An algorithm is developed to annotate the data by using a finite-state approach. A brief highlight of how the system is implemented shows the first step of developing a transliteration tool to transliterate the Nepali text into Roman text. Each word in a sentence is matched to find the verb word, and tagging is performed by following the grammatical rules and categories. The suffix of the word is extracted from the word by finding the patterns using regular expression modules. This system is developed by using Python v3.6 and a must-have NLTK toolkit.

In the above figure, it shows the conversion table of Nepali characters to English characters as Python has a feature of encoding text into utf-8 and because of this Unicode strings are written in normal strings. The pseudocodes for each of the modules are discussed below:

Algorithm: Transliteration

Input: Nepali text
Output: Transliterated text in Roman

Step 1 Start
Step 2 Read Nepali text.
Step 3 Split Nepali text where space is found and store it in *word [].*
Step 4 For each word in *word []*, store the word in *char.*

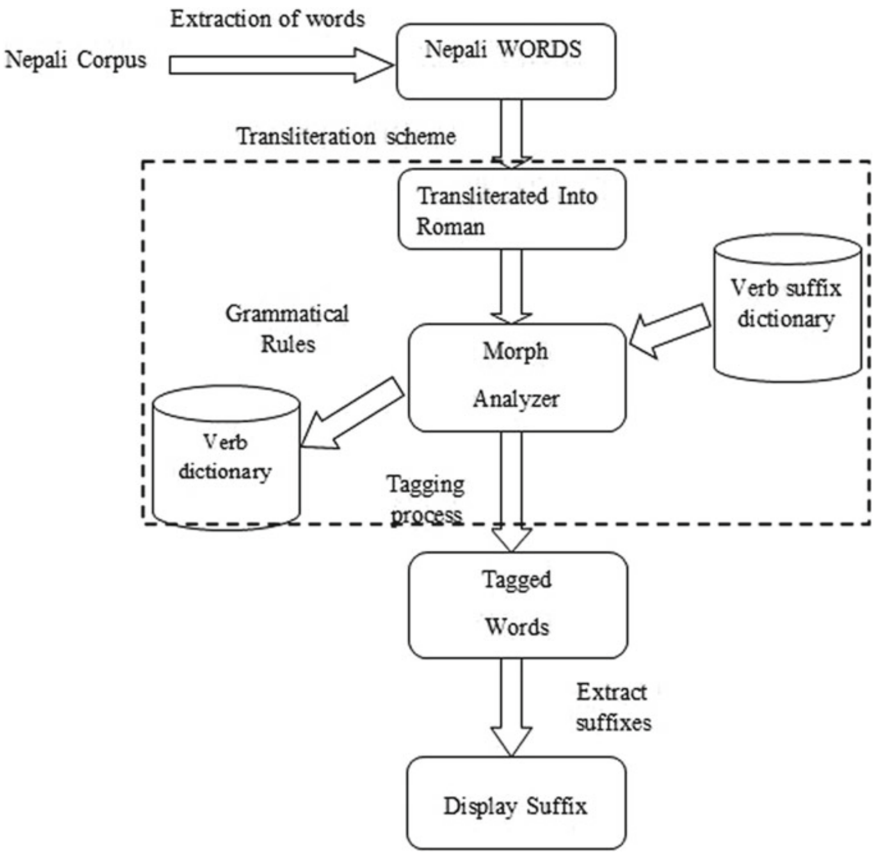


Fig. 1 Block diagram of morph analyzer

Step 5 Check if *char* is a Nepali character, match with Devanagari character.
Step 6 Check if the character is consonants, then the transliteration is dependent upon various factors of nukta, virama, and vowel sign.
Step 7 Select the corresponding English term for the *char* from the conversion dictionary and store in *trans*.
Step 8 Concatenate *trans* to get the transliterated text in *curr*.
Step 9 Stop

Pseudo code: Devanagari to Latin

```
BEGIN
1.  If matches char with consonants, then:
        1.1  nextchar:= word [(index + 1) % len(word)]   //Check the next char
        1.2  if nextchar then:
                1.2.1    If match with nukta then:
                            1.2.1.1    cons:= char + nextchar
                            1.2.1.2    nukta_present:= 1
                1.2.2    else:
                            1.2.2.1    cons:= char
                            1.2.2.2    nukta_present = 0
                1.2.3    End if
                1.2.4    If nukta_present then:
                            1.2.4.1    nextchar:= word [(index + 2) %
                                       len(word)]
                         // if nextchar = vowel sign, virama, convert
                         consonant without "a" else if not then ass "a" to
                         consonant

                1.2.5    If nextchar matches vowelsigns or virama then:
                            1.2.5.1    trans:= conversiontable.get(cons,'')
                            1.2.5.2    curr:= curr + trans
                1.2.6    Else:
                            1.2.6.1    trans:= conversiontable.get(cons,'')
                            1.2.6.2    trans:= trans + "a"
                            1.2.6.3    curr:= curr  + trans
                1.2.7    End if
        1.3  End if
2.  Else:
        2.1  trans:= conversiontable.get(char,'')
        2.2  curr:= curr + trans
3.  End if
        END
```

Pseudocode: Getting the result

```
BEGIN

1.  For each word of main corpus

    1.1  latin_output:= deva_to_latin(word)   //Call the method to translate the text
         with parameter word.
    1.2  All_words.append(latin_output)
    1.3  Joined_all_words:= ‘‘.join(all_words)
```

2. End of for loop

END

Pseudocode: Convert the text BEGIN

1. For each word in line,
   1.1 Call method devanagari to latin, lat_op := deva_to_latn(word)
   1.2 Append to list, words.append(lat_op)
   1.3 Join as string, joined_words := “.join(words)

END

Algorithm: Finite-State Grammar

Pseudocode for detecting verbs and suffix

BEGIN
1. For each lineinnepali_sentence
   1.1 Print the sentence in Nepali
   1.2 Convert Nepali sentence to English, latin := convert(line)
   1.3 Print "Detected verbs"
   1.4 Roman_tag_word := word_tokenize(latin)
   1.5 Replace character Halanta in line, w := re.sub(r'\।',",line)
   1.6 w := word_tokenize(w)
   1.7 For each item in w
      1.7.1 If item present in v_tag dictionary, then:
      Print item and tag the word
      Append in a list, detect_verbs.append(item)
      1.7.2 End If
      1.7.3 For each word in detect_verbs
      Print "suffix for the detected word"
      If re.findall (r'^.*(यो|एको|नु|एका|न|ए|ने) $',w)
      then:
      Print the suffix
      Elseif re.findall (r'^.*(◌ाउने) $',w) then
      Print the suffix
      Elseif re.findall (r'^.*(◌ाउन) $',w) then
      Print the suffix
      Elseif re.findall (r'^.*(◌ेको|◌ेका) $',w) then
      Print the suffix
      Elseif re.findall (r'^.*(न्छु|न्छ|छु) $',w) then
      Print the suffix
      Elseif if re.findall (r'^.*(◌ेर|एर) $',w) then
      Print the suffix

```
                    Elseif re.findall (r'^.*(ा) $',w) then
                                  Print the suffix
                    Elseif re.findall(r'^.*(छि) ',w) then
                                  Print the suffix
                    Elseif re.findall(r'^.*(न्छु) $',w) then
                                  Print the suffix
                    Elseif re.findall(r'^.*(ाउँदै) $',w) then
                                  Print the suffix
                    Elseif re.findall(r'^.*(िएका) $',w) then
                                  Print the suffix
                    Else
                                  Print "Not Found"
                    End If… Elseif... Else
            1.7.4   End of For loop
            1.7.5   For each element in roman_word
                    If element present in r_tag then
                            Print element and tag
                    End If
            1.7.6   End of for loop
      1.8  End of for loop
2.  End of for loop
END
```

5 Results and Discussions

The developed analyzer has been tested in various ways. The corpus or data set was fetched from the files, the tag sets were stored in a dictionary, and also the tagging of the word was fetched and matched from the dictionary stored in a file.

Transliteration See Table 2.

Table 2 Testing of transliteration

Module	Input	Output	Remarks
Transliterator	Nepali word (only consonants)	Transliterated word in Roman	Successfully transliterated into Roman
	Nepali word (consonant + Vowel)	Transliterated word in Roman	Successfully transliterated into Roman
	Nepali word (consonant + vowel + matra)	Transliterated word in Roman	Successfully transliterated into Roman
	Nepali word containing conjunct consonants	Transliterated word in Roman	Successfully transliterated into Roman

5.1 Finite-State Grammar

This section contains the snapshots and discussions of different results by the developed system that was achieved from the system (Table 3).

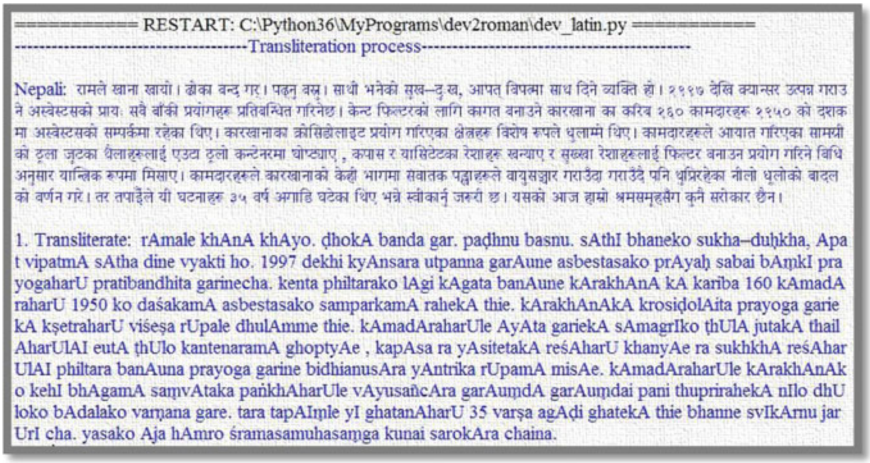
Figure 2 shows the output of Nepali text transliterated into English words by the output of the transliteration tool, and the transliterated text is stored in a file for further processing of the text with POS tagging and suffix extraction. Transliteration is done by using the algorithm given in Sect. 4.

Figure 3 shows the dictionary for all the verb words that is used for tagging, where the words are matched and found in the sentence. This dictionary is manually collected which can be used for further collection and can be implemented in this research work.

In Fig. 4, the tagged verbs are shown in both languages that is available at the verb tag dictionary. From a sentence, each word is checked whether a verb is present and then its respective tags along with the verb word is displayed. For example, in the

Table 3 Testing of finite-state grammar

Module	Input	Suffix checking	Output	Remarks
Finite-state grammar	Word in Roman	Suffix present at end	Tagged word	Successfully tagging for each suffix present
	Word in Roman	Suffix present in middle	Not tagged	Successfully detecting suffix in middle
	Word in Roman	For more than one suffix	Tagged word	Successfully tagging



```
=========== RESTART: C:\Python36\MyPrograms\dev2roman\dev_latin.py ===========
----------------------------------Transliteration process------------------------------------------

Nepali: रामले खाना खायो। ढोका बन्द गर। पढ्नु बस्नु। साथी भनेको सुख–दुःख, आपत् विपत्मा साथ दिने व्यक्ति हो। १९९७ देखि क्यान्सर उत्पन्न गराउने अस्वेस्टसको प्रायः सबै बाँकी प्रयोगहरू प्रतिबन्धित गरिनेछ। केन्ट फिल्टरको लागि कागत बनाउने कारखाना का करिब १६० कामदारहरू १९५० को दशकमा अस्वेस्टसको सम्पर्कमा रहेका थिए। कारखानाका क्रोसिडोलाइट प्रयोग गरिएका क्षेत्रहरू विशेष रूपले धुलाम्मे थिए। कामदारहरूले आयात गरिएका सामग्रीको ठूला जुटका थैलाहरूलाई एउटा ठूलो कन्टेनरमा घोप्ट्याए , कपास र यासिटेटका रेशाहरू खन्याए र सुख्खा रेशाहरूलाई फिल्टर बनाउन प्रयोग गरिने विधि अनुसार यान्त्रिक रूपमा मिसाए। कामदारहरूले कारखानाको केही भागमा सवातक पङ्खाहरूले वायुसञ्चार गराउँदा गराउँदै पनि धुप्रिरहेका नीलो धूलोको बादलको वर्णन गरे। तर तपाईंले यी घटनाहरू ३५ वर्ष अगाडि घटेका थिए भन्ने स्वीकार्नु जरुरी छ। यसको आज हाम्रो श्रमसमूहसँग कुनै सरोकार छैन।

1. Transliterate: rAmale khAnA khAyo. ḍhokA banda gar. paḍhnu basnu. sAthI bhaneko sukha–duḥkha, Apat vipatmA sAtha dine vyakti ho. 1997 dekhi kyAnsara utpanna garAune asbestasako prAyaḥ sabai bAṃkI prayogaharU pratibandhita garinecha. kenta philtarako lAgi kAgata banAune kArakhAnA kA kariba 160 kAmadAraharU 1950 ko daśakamA asbestasako samparkamA rahekA thie. kArakhAnAkA krosiḍolAita prayoga garie kA kṣetraharU viśeṣa rUpale dhulAmme thie. kAmadAraharUle AyAta gariekA sAmagrIko ṭhUlA jutakA thailAharUlAI eutA ṭhUlo kantenaramA ghoptyAe , kapAsa ra yAsitetakA reśAharU khanyAe ra sukhkhA reśAharUlAI philtara banAuna prayoga garine bidhianusAra yAntrika rUpamA misAe. kAmadAraharUle kArakhAnAko kehI bhAgamA saṃvAtaka paṅkhAharUle vAyusañcAra garAuṃdA garAuṃdai pani thuprirahekA nIlo dhUloko bAdalako varṇana gare. tara tapAIṃle yI ghatanAharU 35 varṣa agAḍi ghatekA thie bhanne svIkArnu jarUrI cha. yasako Aja hAmro śramasamuhasaṃga kunai sarokAra chaina.
```

Fig. 2 Transliterated Nepali to English text

Verbs:
khAyo:VBF
dine:VBNE
basnu:VBI
gar:VCN
gare:VBF
garne:VBNE
garna:VBI
gardai:VBO
gardacha:VBF
gareko:VBKO
garera:VBO
garinecha:VBF
garieko:VBKO
gariekA:VBKO
garAune:VBNE
garirahekA:VBKO
garAuṃdA:VBO

खायो:VBF
जान्छु:VBF
दिने:VBNE
सुत्नु:VBI
बस्नु:VBI
पढ्नु:VBI
पर्याउने:VBNE
वस्ने:VBNE
भनिन्छ:VBF
राख्नुपर्दछ:
लागेर:VBO
लिएर:VBO
लेखिछ:VBF
हुनपर्छ:VBF
वनेको:VBKO
विगर्ने:VBNE
मिलेर:VBO
बस्नुपर्छ:VBF
पुर्याउनुपर्छ:VBF
किन्न:VBI
आउनेछु:VBF
गर्:VCN
गरे:VBF

Fig. 3 Verb dictionary (Roman and Nepali)

word ‘khAyo’, suffix present is ‘yo’ and the tag for the word is ‘VBF’. This shows that the action was performed as in simple past tense where it represents third person on speech.

Figure 5 shows the suffix extraction from the word by using the production rules from the suffix dictionary. The suffixes were matched from the dictionary with the tokens, using the regression method ‘*findall ()*’ to extract the suffix from the word and extraction was done successfully.

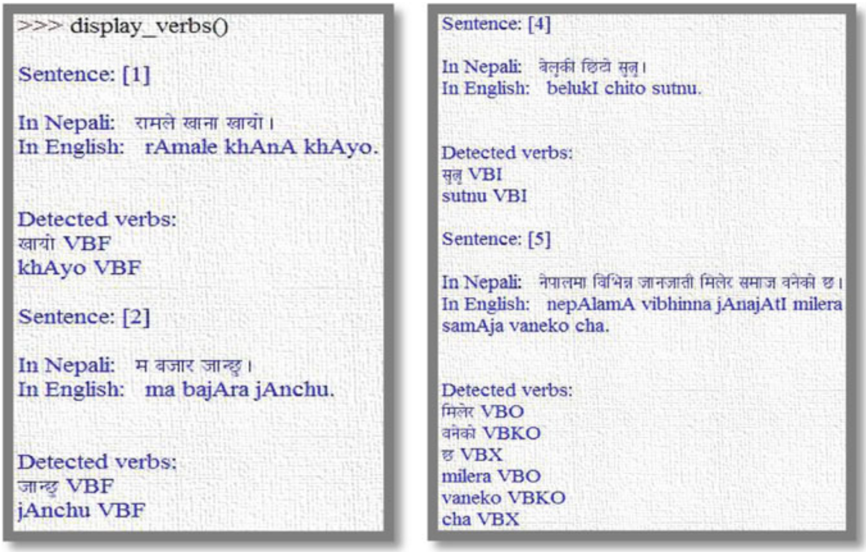


Fig. 4 Tagging of verbs

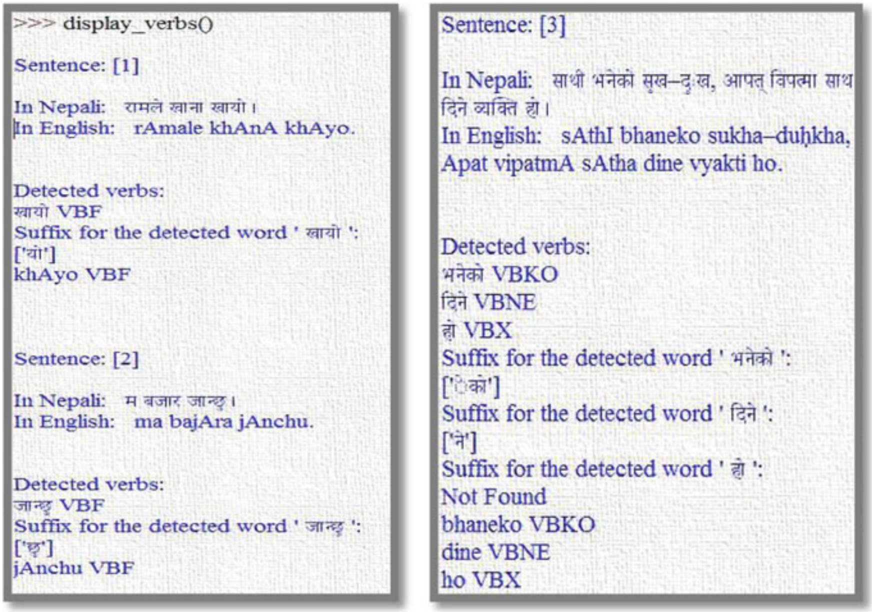


Fig. 5 Extract suffix from word

6 Summary and Conclusions

In this research work, a morph analyzer for Nepali verbs is developed which can be said as a first step toward the development of a successful morph analyzer for Nepali language. To get accurate or in sequence result, various modules are developed to perform the task. The verb of Nepali language is considered with various other grammatical categories. The first task is initiated by using transliteration tool to translate the corpus which was manually written as a prototype to transliterate Nepali text to Roman text. A finite-state grammar approach is used to search and match the words from the dictionary. The word found as a verb based on the rule-based system from each sentence is tagged as a verb showing its category depending on the suffix of the word. The suffix of the word is matched using the production rule, and where the match occurs, the suffix is displayed to the viewer.

This system is developed only for tagging the Nepali verbs with its defined categories and reading and extracting the suffix from the word. With a successful tagging of the words and suffixes, it also has a limitation that the developed system cannot extract the compound verbs that are formed by combining more than two verb words. The verb tagging is not based on contents, so the words are freely tagged and displayed to the viewer.

This research work can be further implemented at a much larger scale by studying the inflections of various other categories found in Nepali verb language. The study can also extend based on the context of the sentence. Once a fully developed morph analyzer is completed, the problem with suffixes as mentioned can be solved by applying the rule-based system.

References

1. Python: https://www.tutorialspoint.com/python/index.htm. Accessed 08 Mar 2018
2. Bhat, S.M., Rai, R.: Building morphological analyser for Nepali. J. Mod. Lang. **22**, 45–58 (2012)
3. Basnet, S.B., Pandey, S.B.: Morphological Analysis of Verbs in Nepali, Nepali Linguistics, vol. 24, Nov 2009. ISSN 0259-1006
4. Choden, U., Lepcha, N.: Morph analyzer for nepali adjectives. Masters thesis, Submitted to Sikkim Manipal Institute of Technology (2016)
5. Chettri, I., Dey, G.: A morph analyzer for Nepali nominals. Masters thesis, Submitted to Sikkim Manipal Institute of Technology (2015)
6. Yajnik, A.: Part of speech tagging using statistical approach for Nepali text. World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng. **11**(1), 201
7. Grosan, C., Abraham, A.: Rule—Based expert systems, intelligent systems. In: ISRL 17, pp. 149–185
8. Prasain, B.: Computational Analysis of Nepali Basic Verbs (Written Forms), Nepali Linguistics, vol. 23, pp. 262–270, November 2008. ISSN 0259-1006
9. Banerjee, S., Das, D., Bandyopadhyay, S.: Bengali verb subcategorization frame acquisition—A baseline model. In: ACLIJCNLP-2009, ALR-7 Workshop, pp. 76–83. Suntec, Singapore (2009)

10. Vikram, S.: Morphology: Indian languages and European languages. Int. J. Sci. Res. Publ. **3**(6) (2013)
11. Saranya, S.K.: Morpphological Analyzer for Malayalam Verbs—A Research work Report. www.pdfcreat.com/computer-malayalam-0.html (2008)

A Proposal of Rule-Based Hybrid Intrusion Detection System Through Analysis of Rule-Based Supervised Classifiers

Ranjit Panigrahi, Samarjeet Borah, and Debahuti Mishra

Abstract Data mining techniques are commonly used for designing intrusion detection systems. The rule-based supervised classifiers play a prominent role in the intrusion detection process and infact, empower the detectors for quick discovery of intrusions in a network of computing devices. However, the design architecture of these classifiers has a significant impact on the speed and accuracy of the detection process. This paper analyzes various rule-based classifiers for designing effective intrusion detection systems. The rule-based classifiers are explored extensively on the basis of detection accuracy, false-positive rate, and ROC value. Three widely used intrusion datasets such as NSL-KDD, ISCXIDS2012, and CICIDS2017 datasets have been considered for effective analysis of rule-based classifiers. Subsequently, the article proposed an architecture considering the best classifier for designing intrusion detection systems.

Keywords Rule-based · Classifiers · NSL-KDD · ISCXIDS2012 · CICIDS2017 · Intrusion · Intrusion detection

1 Introduction

The exponential growth of the World Wide Web (WWW) facilitates the use of effective connectivity and seamless access to information and data. This uncontrolled growth of WWW also has a negative effect on the computing resources on which

R. Panigrahi (✉) · S. Borah
Department of Computer Applications, Sikkim Manipal Institute of Technology, Sikkim Manipal University, Majhitar, India
e-mail: ranjit.panigrahi@gmail.com

S. Borah
e-mail: samarjeetborah@gmail.com

D. Mishra
Department of Computer Science & Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha 751030, India
e-mail: mishradebahuti@gmail.com

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_63

the users rely on. The scattered data and the pattern of accessing such data provide excellent scope for intruders to hack and steal confidential information from the users' computing environments. The intrusion detection system plays a prominent role to counter such illegitimate access of data. The IDSs understands the patterns of the intruders and alert the administrator of taking preventive mechanisms. The intelligence of the IDS detectors comes with the supervised classifiers, which are the basis for almost all signature-based IDS.

The rule-based classifiers are a group of classifiers, that uses general if…else… approach to detect patterns. These classifiers are popular because of the simplicity and fast to operate. Another reason behind the popularity of rule-based approach is its ability to formulate and detect specific attacks in detail [1], which guarantee to detect known attacks with a very little number of false positives. Therefore rule-based classifier approaches are explored extensively among the researchers for designing intrusion detection systems [1–4].

In this article, twelve rule-based classifiers have been analyzed using three widely used intrusion datasets, viz., NSL-KDD, ISCXIDS2012, CICIDS2017, provided by the Canadian Institute of Cybersecurity (CIC). These datasets are recent in nature and represent real-world network attacks. Moreover, all three datasets suffer a high-class imbalance ratio. This provides an opportunity to judge the performance of the classifiers in such a typical environment. In the latter stage, the best rule-based classifier of the analysis has been considered to design an intrusion detection system subsequently. The rest of this article are as follows: Sect. 2 highlights related research works pertaining to rule-based classifiers. Section 3, analyzes twelve rule-based classifiers based on the parameters—false-positive rate, accuracy and ROC value. Section 4 represents an architecture of a rule-based intrusion detection system considering the best classifier evolved in Sect. 2. Finally, the article is concluded at Sect. 5.

2 Literature Reviews

Analysis of supervised classifies has been undertaken by many researchers either for validating a new approach [5, 6] or a systematic qualitative analysis for a specific environment [7–9]. Most of these analysis focuses on a few performance measures such as accuracy and misclassification rate. Moreover, a large section of the analysis is evaluated on a balanced training set. Even though many other analysis reveals the optimum classifiers in the class imbalanced environment but these analyses are fail to explain the potential of classifiers by merely considering classification accuracy. Supervised classifies such as Support Vector Machine (SVM), decision tree and K-Nearest Neighbour has been analyzed by Aksu et al. [10]. The authors conducted the study on the top of fisher score feature selection. Moreover, the analysis was conducted on binary class instances of CICIDS2017 dataset. In the analysis, KNN emerged as the best classifier. However, the true performance of KNN is not known in a class imbalance issue. Panda and Patra [11] investigated various rule-based classifiers on KDDCUP'99 dataset. A detection rate of 0.993 has been achieved for rule-based Decision Table (DT) while segregating normal instances form attack instances. As a potential shortcoming, the analysis doesn't address class imbalance problem. The same dataset has also been explored by Garg and Khurana [12] through multiple supervised classifiers, where the Rotation Forest (RF) emerged as the best classifier. Garrett" Ranking Procedure has been employed to rank classifiers according to the performance observation to get the best classifier. Kshirsagar and Joshi [13] presented a study of rule-based classification methods for intrusion detection systems. The accuracy rate of classifiers is the only parameter considered for evaluating intrusion detection. Ripper rule learner and formal concept analysis have been explored by Kotelnikov and Milov [14], where the formal concept analysis has been emerged as the best mechanism with the average precision rate of 86.8%.

3 Analysis of Rule-Based Classifiers

Many articles analyzed rule-based classifiers considering varieties of parameters, datasets and environments. However, most of those approaches have not addressed one or more of the potential areas

- Exclusive analysis of rule-based classifiers.
- Class imbalance issue.
- Classifiers are analyzed considering only few performance measures.

In this section, 12 state-of-the-art rule-based classifiers have been analyzed considering three recent intrusion detection datasets, viz., NSL-KDD [15], ISCXIDS2012 [16] and CICIDS2017 [17, 18] provided by Canadian Institute of Cybersecurity. These datasets are highly class imbalance in nature [19, 20]. Therefore, we have considered ROC value analysis in order to come across the best classifier to design an intrusion detection system. The rule-based classifiers considered for analysis are outlined in Table 1.

In order to evaluate the classifiers outlined in Table 1, a reasonable number of samples from CIC datasets have been considered randomly without altering any attributes. This ensures the sample set will retain the properties of the original dataset. The sample set with training and test instances are outlined in Table 2. Here 66 and 34% of the sample size has been considered as training and testing instances.

Once we have an adequate number of samples then the classifiers are trained on training instances and the testing instances are assigned to trained classifiers for classification. Upon the classification, the performance observations are tabulated for analysis. We have considered classification accuracy, false-positive rate and ROC value for analysis. The classification outcomes are tabulated in Tables 3 and 4, where Table 3 represents classification accuracy, Table 4 represents false-positive rate and Table 5 represents ROC value of the classifiers.

Table 1 Rule-based classifiers

Classifier name	Description
Conjunctive Rule (CR) [21]	Based on a single conjunctive rule learner. It can predict nominal as well as numeric class instances
Decision Table (DTBL) [22]	It builds and uses a simple Decision Table majority classifier
Decision Table and Naïve Bayes (DTNB) hybrid [23]	It employs a semi-Naïve Bayesian ranking method, where Naïve Bayes is combined with induction of decision tables
Repeated Incremental Pruning to Produce Error Reduction (RIPPER) [24]	It is an extension of IREP classifier. Propositional rule learner is implemented
Fuzzy Unordered Rule Induction Algorithm (FURIA) [25]	Extension of RIPPER algorithm. Instead of using conventional rules and unordered rule sets, it learns fuzzy rules
MODLEM [26, 27]	MODLEM algorithm uses rough set theory to handle inconsistent examples and computes a single local covering for each approximation of the concept
Nearest Neighbour with Generalization (NNGE) [28, 29]	It uses nearest-neighbour-like algorithm using non-nested generalized exemplars to predict unknown instances
Ordinal Learning Method (OLM) [30]	Automatically generates symbolic rule-bases from examples
OneR [31]	Builds 1R algorithm that uses minimum-error attribute for prediction
PART [32]	It employs separate and conquer principle by building a partial C4.5 decision tree where it makes the best leaf into a rule induction
RIpple-DOwn Rule Learner (RIDOR) [33]	It produces a default rule and exceptions for it with the least amount of error rate followed by best exceptions for each exception and iterates until pure case achieved
Rough Set (ROUGHS) [34]	Classifier with rule induction based on rough sets

Table 2 Sample size, training and testing instance ratio of CIC intrusion datasets

Datasets	Sample size	Training instances	Testing instances
NSL-KDD	7781	5135	2646
ISCXIDS2012	5494	3626	1868
CICIDS2017	8917	5885	3032

Table 3 Classification accuracy of rule-based classifiers

Rule-based classifiers	NSL-KDD	ISCXIDS2012	CICIDS2017
CR	49.282	82.923	38.358
DTBL	94.067	94.968	99.67
DTNB	94.482	97.216	**99.868**
RIPPER	96.636	97.323	99.637
FURIA	95.881	**97.912**	99.703
MODLEM	95.011	97.216	99.604
NNGE	94.52	96.842	99.835
OLM	53.061	72.056	72.131
ONER	78.949	91.756	98.153
PART	**97.241**	**97.912**	99.769
RIDOR	95.918	97.216	99.67
ROUGHS	96.863	97.056	99.769

Bold indicates the highest performance observed for the corresponding classifier.

Table 4 False-positive rate of rule-based classifiers

Rule-based classifiers	NSL-KDD	ISCXIDS2012	CICIDS2017
CR	0.1000895	0.00222399	0.00741955
DTBL	0.00981618	0.00064155	0.0000393
DTNB	0.00717198	0.00034845	0.0000155
RIPPER	0.00172011	0.00028732	0.0000369
FURIA	0.0035028	0.00024523	0.0000329
MODLEM	0.00169279	0.00026806	0.0000364
NNGE	0.00158963	0.00028768	**0.0000144**
OLM	0.01325267	0.00251141	0.00241638
ONER	0.0058557	0.00073615	0.00015914
PART	0.00075095	**0.00018491**	0.0000197
RIDOR	0.00095148	0.00023286	0.0000266
ROUGHS	**0.00065496**	0.00023543	0.0000185

Bold indicates the highest performance observed for the corresponding classifier.

It is observed from Table 3 that PART shows highest amount of accuracy while tested on both NSL-KDD and ISCXIDS2012 dataset, whereas DTNB shows a maximum accuracy of 99.868%.

Considering false-positive rate, the rule-based classifiers reveal a diversified result. For the three datasets three different classifiers rough set, PART and NNGE shows the least amount of false alarms. The PART classifier, which revealed the highest

Table 5 ROC value of rule-based classifiers

Rule-based classifiers	NSL-KDD	ISCXIDS2012	CICIDS2017
CR	0.78	0.83	0.78
DTBL	**0.99**	0.97	**1**
DTNB	**0.99**	**0.99**	**1**
RIPPER	**0.99**	**0.99**	**1**
FURIA	**0.99**	0.98	**1**
MODLEM	0.97	0.97	**1**
NNGE	0.97	0.97	**1**
OLM	0.71	0.72	0.84
ONER	0.86	0.92	0.99
PART	**0.99**	**0.99**	**1**
RIDOR	0.97	0.97	**1**
ROUGHS	0.98	0.97	**1**

Bold indicates the highest performance observed for the corresponding classifier.

amount of accuracy for the NSL-KDD dataset loses its position with a very marginal difference of false alarm. However, PART retains its position as the best classifier for ISCXIDS2012 by maintaining least false alarms. Similarly, for CICIDS2017 the Nearest Neighbour with Generalization shows a very little false alarm of 0.0000144.

Subsequently, for ROC values the results are quite interesting. A large chunk of rule-based classifiers shows ROC value approximately equal to 1 both for NSL-KDD and ISCXIDS2012. On the other hand, nine classifiers reveal ROC value of 1 for CICIDS2017 dataset.

Now at this point to identify the best classifier for designing an effective intrusion detection system for class imbalance scenarios, we have focused on CICIDS2017 dataset. It is because this dataset claims to fulfil all the 11 criteria of a true IDS as described by Gharib et al. [35]. For CICIDS2017, the hybrid of Decision Table and Naïve Bayes (DTNB) seems to be an ideal classifier. The Decision Table and Naïve Bayes hybrid outperform all other classifiers in two out of three performance parameters. The DTNB shows highest accuracy and ROC value for CICIDS2017 dataset. Moreover, the ROC value of 1 signifies DTNB excel well in high-class imbalance situations. Therefore, DTNB can be considered as a base learner to design a new intrusion detection system.

4 An IDS Framework Proposal

In this section, an attempt has been made to sketch an IDS based on the analysis conducted in Sect. 3. The analysis revealed that a Decision Tree and Naïve Bayes

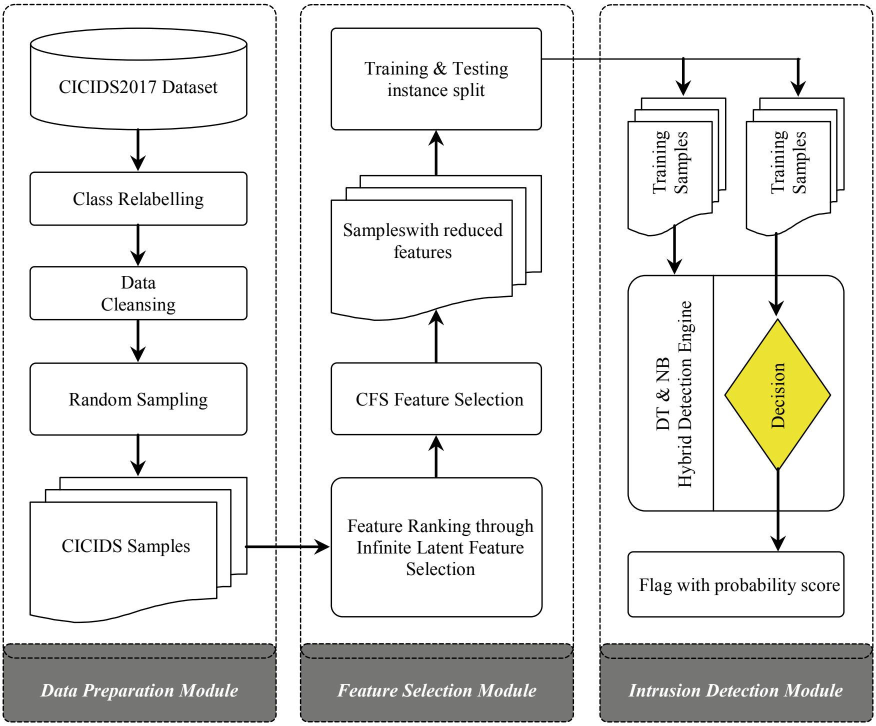


Fig. 1 A proposed architecture of the intrusion detection system

is an effective IDS for CICIDS2017 dataset. Since CICIDS2017 dataset is the most recent IDS dataset, therefore we have considered this dataset to design an IDS. A schematic diagram of the proposed detection model has been presented in Fig. 1.

The proposed IDS can be composed of the following modules.

4.1 Data Preparation Module (DPM)

The CICIDS2017 dataset has been analyzed briefly by many researchers [18, 36] due to its significant impact on IDS. In our previous research work, the dataset has been balanced using class relabelling [19]. In this regard, the attacked classes can be grouped to form new attack classes; thus, improving class imbalance ratio. A detailed guideline can be found at [19]. The guideline can be followed to transform the original dataset into a balanced dataset. Once the dataset has been balanced, a reasonable number of attacks and benign samples can be selected for the next module of the proposed IDS. Before generating samples, any duplicate instances must be removed.

Once the data is free from duplicate and missing value instances, and attacks are relabelled then the appropriate number of class-wise random samples need to be generated. The attack-wise sample ensures a smaller number of samples having equal weightage for each class; thus, reducing class imbalance significantly.

4.2 Feature Selection Module (FSM)

The CICIDS2017 dataset contains 80 features. Out of these features many contributed well to the classification process and many have little impact. Many features are redundant due to similarity in feature distances. Therefore, a suitable feature selection seems to improve attack detection. Moreover, a detection model built on few features will have a fast response to attack as compared to a detection model trained on all the 80 features.

Feature ranking plays a vital role in the classification process [37–39]. These approaches ranked the features according to the feature weight. Similarly, feature selection procedure ensures selection of the best feature subset for optimum classification. In this proposed architecture a combination of feature ranking and feature selection method ensures the selection of best feature subset arranged according to their feature weights. For feature ranking, Infinite Latent Feature Selection (ILFS) [40] and for feature selection, Correlation-based Feature Selection (CFS) has been deployed. At the end of this module, a sample of CICIDS2017 instances with reduced features will be obtained. The reduced features sample can be divided into two sets of training and testing instances.

4.3 Intrusion Detection Module (IDM)

The IDM module solely involves in the detection process. This module employs Naïve Bayes and rule-based Decision Table as a hybrid detector. The hybrid module can be trained on the training samples obtained by FSM module. Finally, the test instances can be used to evaluate the trained hybrid detector. The detection model expected to achieve better accuracy with a lower false alarm rate as compared to state-of-the-art detection models proposed on CICIDS2017 dataset.

5 Conclusion

In this article twelve rule-based classifiers have been analyzed in a high-class imbalance environment for ascertaining the best classifier, which can act as the base learner of any intrusion detection system. In order to create a class imbalance scenario, three-class imbalance intrusion datasets have been considered for evaluation. The classification accuracy, false alarm rate and ROC value are used to analyze the classifiers performance. The PART classifier emerged as the best classifier for NSL-KDD and ISCXIDS2012 datasets. Similarly, Decision Table with Naïve Bayes (DTNB) hybrid outperforms other classifiers in CICIDS2017 dataset. The outcome of the analysis has been used to sketch an intrusion detection architecture through Naïve Bayes and Decision Table hybrid for CICIDS2017 dataset. As future work, the architecture can be used to develop an intrusion detection system using CICIDS2017 dataset.

References

1. Li, L., Yang, D.Z., Shen, F.C.: A novel rule-based intrusion detection system using data mining. In: 2010 3rd International Conference on Computer Science and Information Technology, vol. 6, pp. 169–172. IEEE (2010)
2. Duffield, N., Haffner, P., Krishnamurthy, B., Ringberg, H.A. U.S. Patent No. 9,258,217. U.S. Patent and Trademark Office, Washington, DC (2016)
3. Turner, C., Jeremiah, R., Richards, D., Joseph, A.: A rule status monitoring algorithm for rule-based intrusion detection and prevention systems. Procedia Comput. Sci. **95**, 361–368 (2016)
4. Patel, S.K., Sonker, A.: Rule-based network intrusion detection system for port scanning with efficient port scan detection rules using snort. Int. J. Future Gener. Commun. Netw. **9**(6), 339–350 (2016)
5. Seetha, H., Saravanan, R.: On improving the generalization of SVM classifier. In: International Conference on Information Processing, pp. 11–20. Springer, Berlin, Heidelberg (2011)
6. Adnan, M.N., Islam, M.Z.: Forest PA: constructing a decision forest by penalizing attributes used in previous trees. Expert Syst. Appl. **89**, 389–403 (2017)
7. Portal, G.M., Ghersi, A.G., Juárez, P.S., Valenzuela, R.G.: Comparative analysis of supervised classifiers for classification of musical notes on mobile based applications. In: Proceedings of the 2nd International Conference on Vision, Image and Signal Processing, p. 21. ACM (2018)
8. Amancio, D.R., Comin, C.H., Casanova, D., Travieso, G., Bruno, O.M., Rodrigues, F.A., da Fontoura Costa, L.: A systematic comparison of supervised classifiers. PLoS ONE **9**(4), e94137 (2014)
9. Mushtaq, Z., Yaqub, A., Hassan, A., Su, S.F.: Performance analysis of supervised classifiers using PCA based techniques on breast cancer. In: 2019 International Conference on Engineering and Emerging Technologies (ICEET), pp. 1–6. IEEE (2019)
10. Aksu, D., Üstebay, S., Aydin, M.A., Atmaca, T.: Intrusion detection with comparative analysis of supervised learning techniques and fisher score feature selection algorithm. In: International Symposium on Computer and Information Sciences, pp. 141–149. Springer, Cham (2018)
11. Panda, M., Patra, M.R.: Ensembling rule based classifiers for detecting network intrusions. In: 2009 International Conference on Advances in Recent Technologies in Communication and Computing, pp. 19–22. IEEE (2009)
12. Garg, T., Khurana, S.S.: Comparison of classification techniques for intrusion detection dataset using WEKA. In: International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), pp. 1–5. IEEE (2014)
13. Kshirsagar, V., Joshi, M.S.: Rule based classifier Models for intrusion detection system. Int. J. Comput. Sci. Inf. Technol. **7**(1), 367–370 (2016)
14. Kotelnikov, E.V., Milov, V.R.: Comparison of rule induction, decision trees and formal concept analysis approaches for classification. J. Phys.: Conf. Ser. **1015**(3), 032068 (2018) (IOP Publishing)

15. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6. IEEE (2009)
16. Shiravi, A., Shiravi, H., Tavallaee, M., Ghorbani, A.A.: Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput. Secur. **31**(3), 357–374 (2012)
17. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSP, pp. 108–116 (2018)
18. Sharafaldin, I., Habibi, Lashkari, A., Ghorbani, A.A.: A detailed analysis of the CICIDS2017 data set. In: Mori, P., Furnell, S., Camp, O. (eds.) Information Systems Security and Privacy. ICISSP 2018. Communications in Computer and Information Science, vol. 977. Springer, Cham
19. Panigrahi, R., Borah, S.: A detailed analysis of CICIDS2017 dataset for designing intrusion detection systems. Int. J. Eng. Techno. **7**(3.24), 479–482 (2018)
20. Panigrahi, R., Borah, S.: Rank allocation to J48 group of decision tree classifiers using binary and multiclass intrusion detection datasets. Procedia Comput. Sci. **132**, 323–332 (2018)
21. Rabi, R., Joanisse, M.F., Zhu, T., Minda, J.P.: Cognitive changes in conjunctive rule-based category learning: An ERP approach. Cogn. Affect. Behav. Neurosci. **18**(5), 1034–1048 (2018)
22. Kohavi, R.: The power of decision tables. In: European Conference on Machine Learning, pp. 174–189. Springer, Berlin, Heidelberg (1995)
23. Hall, M.A., Frank, E.: Combining Naive Bayes and decision tables. In: FLAIRS Conference, vol. 2118, pp. 318–319 (2008)
24. Cohen, W.W.: Fast effective rule induction. In: Machine Learning Proceedings 1995, pp. 115–123. Morgan Kaufmann (1995)
25. Hühn, J., Hüllermeier, E.: FURIA: an algorithm for unordered fuzzy rule induction. Data Min. Knowl. Disc. **19**(3), 293–319 (2009)
26. Stefanowski, J.: The rough set based rule induction technique for classification problems. In: Proceedings of 6th European Conference on Intelligent Techniques and Soft Computing EUFIT, vol. 98 (1998)
27. Grzymala-Busse, J.W., Stefanowski, J.: Three discretization methods for rule induction. Int. J. Intell. Syst. **16**(1), 29–38 (2001)
28. Martin, B.: Instance-based learning: nearest neighbor with generalization (1995)
29. Roy, S.: Nearest Neighbor with Generalization. Christchurch, New Zealand (2002)
30. Ben-David, A.: Automatic generation of symbolic multiattribute ordinal knowledge-based DSSs: methodology and applications. Decis. Sci. **23**(6), 1357–1372 (1992)
31. Holte, R.C.: Very simple classification rules perform well on most commonly used datasets. Mach. Learn. **11**(1), 63–90 (1993)
32. Frank, E., Witten, I.H.: Generating accurate rule sets without global optimization (1998)
33. Gaines, B.R., Compton, P.: Induction of ripple-down rules applied to modeling large databases. J. Intell. Inf. Syst. **5**(3), 211–228 (1995)
34. Wojna, A., Kowalski, L.: RSESLIB programmer's guide. In: Faculty of Mathematics, Informatics and Mechanics. University of Warsaw (2017)
35. Gharib, A., Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: An evaluation framework for intrusion detection dataset. In: 2016 International Conference on Information Science and Security (ICISS), pp. 1–6. IEEE (2016)
36. Boukhamla, A., Gaviro, J.C.: CICIDS2017 dataset: performance improvements and validation as a robust intrusion detection system testbed
37. Roffo, G., Melzi, S., Cristani, M.: Infinite feature selection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4202–4210 (2015)
38. Gu, Q., Li, Z., Han, J.: Generalized fisher score for feature selection. (2012). arXiv:1202.3725

39. Liu, H., Motoda, H. (eds.): Computational Methods of Feature Selection. CRC Press (2007)
40. Roffo, G., Melzi, S., Castellani, U., Vinciarelli, A.: Infinite latent feature selection: a probabilistic latent graph-based ranking approach. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1398–1406 (2017)

File Encryptor

ASVK Vinayak, Sudeep Ranjan Swain, and Satyabrata Maity

Abstract In today's world, security is one of the major challenges which are faced by the people. For securing our confidential data, encryption is done. Encryption is the process that protects our important information from unwanted people accessing or manipulating it. Cryptography is one of the methods for encryption and decryption of data. With the evolution of human intelligence, cryptographic forms have become more complex in order to have our files and information more secure, and arrays of encryption systems are being used for protecting our data from tampering by various organizations. In this paper, our strategy used is to apply the encryption methods using Advanced Encryption Standard (AES) in order to proceed on both encryption and decryption.

Keywords Cryptography · Advanced Encryption Standard (AES) · Encryption · Decryption

1 Introduction

In today's world, the Internet is playing a vital role in our life. We can communicate in remote areas using the Internet, we can share our documents and important files through the Internet, and also we can protect it from hackers. From the encryption/decryption process, we can achieve security of data and data are secured more. Cryptography is a method for not disclosing the content so that the third party cannot understand the process for stealing data. Cryptography mainly deals with the

A. Vinayak (✉) · S. R. Swain · S. Maity
Computer Science and Engineering, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India
e-mail: Asvkvinayak@gmail.com

S. R. Swain
e-mail: sudeepswain2000@gmail.com

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_64

scrambling of data, such as text and image, and so it hides the data, so that third-party person cannot decode it. Cryptography is originated from the Greek words "cryptos" and "graph". It is the method of converting meaningful data to meaningless data or garbage value and again transforming the data into its original form. It involves two processes—encryption and decryption. Encryption involves the process of changing plain text into ciphertext. Decryption is the reverse process of encryption. Cryptography is one of the concrete pillars which are used for securing files and data.

Have you ever wondered in WhatsApp it displays "messages to this chat and calls are secured with end–end encryption"? It means that our conversations and messages cannot be seen by others because it encrypts our conversations so that the third party cannot understand our messages. WhatsApp uses the Curve25519-based algorithm with elliptic curve Diffie–Hellman (ECDH) key which encrypts our messages most securely. Signal protocol for end–end encryption is designed in such a way that it prevents third parties and WhatsApp from having plain text access to messages or calls which means it is more secure and reliable. Nowadays, we backup our files in Google cloud, and even Google encrypts our data which are stored in cloud so that it will protect our data from cloud data leaks. Even Facebook stores our password in encrypted form so that no one can open our account and leak data.

In cryptography, there are many features like

Authentication: It is a process in which identity is given to someone to access particular resource using the keys.

Confidentiality: It also ensures that nobody understands the message except the one who has the cipher key.

Data Integrity: It is the process in which it ensures that nobody is allowed to alter the transmitted message except the party who is allowed to do so.

Non-repudiation: **It** ensures that neither the sender nor the receiver of the message should be allowed to deny the transmission of the message.

Access Control: It ensures that only the authorized parties are able to access the transmitted message.

2 Related Work

The authors in [1] describe the comparison of different types of algorithms like AES and RSA which is used in wireless intrusion detection system.

The authors in [2] describe the summary of the classification of the types of the encryption techniques, the parameters like encryption ratio, speed, key length, etc., are verified for the algorithms, and the security issues are briefly placed.

The authors in [3] describe the challenges and problems on design and analysis of image encryption schemes summarized to receive the attentions of both designers and attackers for image encryption.

The authors in [4] describe about cryptography and different types of cryptographic algorithms like AES, RSA, DES and 3DES and convey which algorithms are secure and take less memory space.

The authors in [5] describe the paper and conclude a detailed study of different types of algorithms which are used for encryption like AES, DES and 3DES. They conclude that AES is more secure than other algorithms.

3 Proposed Work

In today's scenario, many cryptographic algorithms are available which encrypt data and files. In today's generation, people are busy in their laptops and basically our program depends upon the processor of the system; i.e. if it is fast processor, their time will be consumed less. Also, it depends on the length of the key entered by the user; if it is large, it takes more time for encryption. Nowadays in the market, AES algorithm is comparatively more safe and secure. Here in our module, we have implemented AES algorithm to ensure quite a safe method for encryption and decryption purpose.

Nowadays, everyone uses Microsoft Office for making their office-related projects, maintaining student's records in schools or our bank records which are maintained by banks, etc. But have you ever wondered how secure our information is? Every big organization like banks and government sectors makes its documents encrypted so that it will become more secure and information will not be tampered by third party. Basically in cryptographic techniques, data is altered using some keys which are entered by user so that data cannot be readable and it will make the data more secure so that no one can tamper it or misuse it. In file, Encryptor Text File is encrypted by using XOR with a key which is public or private key. Keys are asymmetric which means that different keys are used for encryption and decryption, which is also known as public key encryption. We can use any type of key for encryption; i.e. it can be numeric or alphanumeric. User can enter two types of keys, i.e. public and private. Receiver uses a private key for decryption, and sender uses public key while sending the files which are confidential.

Advanced Encryption Standard (AES) which was developed by two Belgian cryptographers Joan Daemen and Vincent Rijmen, was used for encrypting the files, which was later published by National Institute of Standards and Technology (NIST). It encrypts and decrypts a data block of 128 bits. It can take 10, 12 or 14 rounds. The key size can vary accordingly like 128,192 or 256 bits depending on the number of rounds. In this paper, we explained AES algorithm is a central base for our project which explains that this program helps to protect our data from hackers. Starting with a functional description of AES, which is changed to some unknown bits which can been named as ciphertext, we use a key with an AES algorithm.

Encrypted text is reversed back with another key to get the original message.

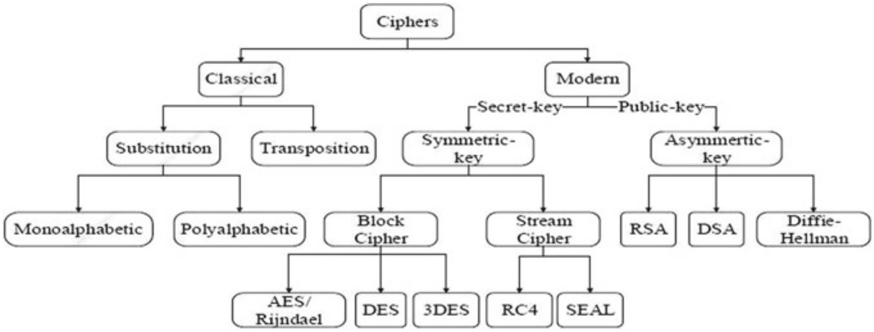


Fig. 1 Different types of algorithms used in cryptography

There are basically four rounds in AES algorithm which are as follows:

1. Substitute bytes: Uses an S-box to perform a byte-by-byte substitution of the block.
2. Shift rows: A simple permutation.
3. Mix column: A substitution method where data in each column from the shift row is multiplied by the algorithm's matrix.
4. Add round key: The key for the processing round is XORed with the data (Fig. 1).

It also explains the process of cryptography and how we can use different algorithms for different types of encryption.

The basic steps which are involved in encryption model are shown below:

- If a sender wants to send a "Password" message to receiver side, then we can do encryption so that third party cannot understand the message.
- Receiver uses a private key for decryption, and sender uses public key while sending the files which are confidential.
- If different keys are used with same algorithm, then we get different types of encrypted message and we can get clarity how secure our files are because if different keys produce same output then it can be decoded easily.

After encryption is over, user will see "Password" as "^&! #%1@#" because the message is "XOR" with the key which was entered by the user.

- From the changed message if we decrypted it, then we can get the original message from the different keys which were entered by the user.
- We humans can see objects with their colour which is a combination of RGB. We can get many colours from mixing three basic colours, RGB.
- We can secure our images with more safety by using image encryption.
- Images are basically the combination of individual pixel (dots) information. Extracting and manipulating the RGB values of each pixel will thus result in changing the original data of the information.
- For security measures, in all the above-mentioned ways of encryption the data is scrambled and when user enters the correct password then only, he/she will get

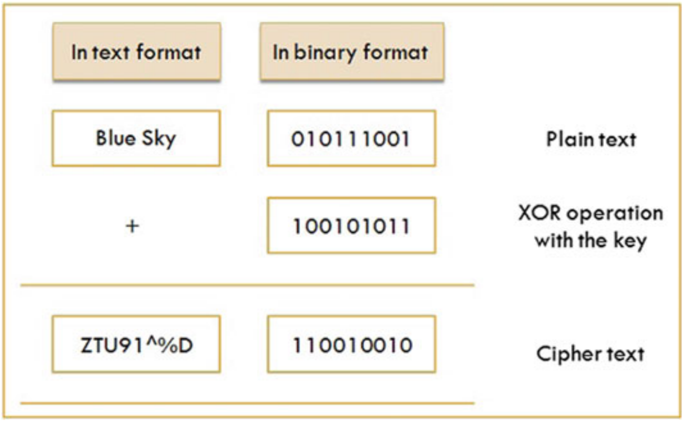


Fig. 2 Process of how plain text is converted into ciphertext

the original data restored; otherwise, data will be in the same scrambled form which will be quite difficult to decode (Fig. 2).

It displays the pictorial representation of how our encryption is done. It also explains how messages which are XORed with keys really work.

Similarly, we have performed image encryption which encrypts our image and it is also more secure.

Image is basically combination of individual pixel (dots) information. When we write that image is of 620 × 480 size, it means that image has 620 pixels in horizontal direction and 480 pixels in vertical direction. So, altogether there are 620 × 480 pixels and each pixel contains some information about image. Pixel extraction is a way in which an image containing a certain number of pixels (row × column) is one by one being extracted as a combination of matrix. Image scrambling is an effective method for providing image security. After the pixels are extracted, then the RGB values of each pixel are changed by increasing or decreasing the colour values. The pixel scrambling is initiated to change the image pixel positions through matrix transform to achieve the visual effect of disorder. This paper presents digital image scrambling based on modified shuffling. The method proposed is a simple but quite a powerful technique. The proposed algorithm and the private key transform drastically the statistical characteristic of original image information, so it increases the difficulty of an unauthorized individual to break the encryption. The simulation results and the performance analysis show that the algorithm has large secret key space, high security, fast scrambling speed and strong robustness.

4 Results and Analysis

Original text:

```
My project name is file encryptor
It encrypts text,images and important files
```

Encrypted text:

```
5c!+%58=9k8(6;¦8?230%w97>♥iqpor¦ant filds

♀;c47>--*>k",#*p8!s2<:2x<4@reptoz
```

And after encrypting if user wants to decrypt, then we have to enter a private key; if private key entered is correct, then the original message will be shown; otherwise, scrambled message will be shown so that if third party decodes it even though they will not get original message.

Let private key be "encryption".

If user enters private key correctly, then message will be

```
My project name is file encryptor
It encrypts text,images and important files
```

If user enters wrong private key, then message will be

```
It encrypts text,images and important files
My project name is file encryptor
```

5 Conclusion and Future Work

This paper is formulated after a detailed study and analysis of AES algorithm which explains a detailed concept about cryptography procedure. Cryptography is mainly based on two prior techniques which involves symmetric and asymmetric accesses. Symmetric access algorithm which uses a symmetric key which lies with both sender and receiver is used for both encryption and decryption, while on the other hand asymmetric algorithms use asymmetric pair of keys for such purpose. So on analysing, we can conclude that AES as in comparison with other algorithms like DES and 3DES is quite secure, authentic and reliable like effectiveness, speed, encryption and decryption time, memory usage.

Future work may include different parameters to improve the encryption ratio and quantum key encryption where quantum nature of atoms protects the data.

References

1. Singh, G., Singla, A., Sandha, K.S.: Cryptography algorithm comparison for security enhancement in wireless intrusion detection system. Int. J. Multidisc. Res. **1**(4), 143–151 (2011)
2. Jeeva, A.L., Palanisamy, V.., Kanagaram, K.: Comparative analysis of performance efficiency and security measures of some encryption. Int. J. Eng. Res. Appl. (IJERA) **2**(3). ISSN: 2248–9622 (2012)
3. Li, C., Zhang, Y., Xie, E.Y.: When an attacker meets a cipher-image in 2018: a year in review. Inf. Secur. Appl. **48** (2019)
4. Singh, G., Supriyaa: Study of encryption algorithms (RSA, DES, 3DES and AES) for information security. Int. J. Comput. Appl. (0975–8887) **67**(19) (2013)
5. Alanazi, H.O., Zaidan, B., Zaidan, A., Jalab, H.A., Shabbir, M., Al-Nabhani, Y.: New comparative study between DES, 3DES and AES within nine factors. J. Comput. **2**(3):152–157 (2010)

Information Security Using Multimedia Steganography

Ayush Kejariwal, Sourav Bera, Shubhanjan Dash, and Satyabrata Maity

Abstract Nowadays, use of data is growing exponentially with the advancement cyber technology. Hence, the security of the data is one of the major areas of concern as a hacker has set to work to crack this data for their cause. Steganography implements the technique for hiding the confidential information into a cover file in order to maintain the security, which is difficult to extract for the third party. The proposed approach includes two different forms of multimedia steganography namely image and video. Video steganography is also an efficient technique for storing a large amount of data. We have used least significant bit (LSB)-based techniques for image and video-based steganography. The degree of steganography can hide a large capacity of data and security provided by it through algorithm used. The results show the effectiveness of our model.

Keywords Multimedia steganography · Stego-key · Discrete cosine transform (DCT) · Cryptography

1 Introduction

Nowadays, information is getting more and more difficult to conceal. It is a lot harder to hide information if someone knows there is some information to look for. The idea behind hiding information and securing it is to not get noticed in the

A. Kejariwal (✉) · S. Bera · S. Dash · S. Maity
Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'o' Anusandhan University, Bhubaneswar, Odisha, India
e-mail: kejayush29@gmail.com

S. Bera
e-mail: sauravberachinmaya3@gmail.com

S. Dash
e-mail: shubhanjan.dash@gmail.com

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_65

first place. Cryptography is the most sought-after method to hide information, but steganography is much more versatile and secure.

Steganography is the method in which data is encoded in a medium. It keeps our information secure and can even be used in combination with other encrypting methods. Even if the encoded message is detected, it is really hard to decode and decipher. It has many applications and promotes secure communication. Different mediums can be used to hide data. It covers various methods and types of steganography and is discussed in depth.

1.1 Essential Features of Steganography Method

- Information Hiding Capacity: Amount of data or information that can be encoded in the image or video without altering its original quality.
- Not noticeable: Information should be hidden in such a way that no one can directly notice or is visible with naked eyes.
- Retainability: The algorithm is having the ability to save the encoded data even after encoding and decoding it.
- Tamper Resistance: If someone bypasses this technique also, they cannot manipulate or delete or read it without the algorithm.

1.2 Steganography and Cryptography

The steganography means hiding or covered, but cryptography refers to altering the secret text to make it unreadable. Steganography was invented for secured undetected communication, but cryptography makes it readable to the targeted recipient who knows to decode the embedded form of a message. In the steganographic method, the framework of the message remains unchanged, but in cryptography, the mainframe of the message is changed to give its disguised form, though cryptography is prevalently used over years, whereas steganography has not got much popularity yet. The degree of hiding the message depends upon the length of the key algorithm, whereas steganography uses its algorithms to make data secured. Steganography has confidentiality and authentication. Spatial domain, embedding, model-based and transform domain are some used algorithms used in steganography. The cryptography uses transposition, block cyphers, substitution and stream. Whereas steganography can be used in various mediums, cryptography can only be applied on a text file. The technique used to detect the presence of the steganography is known as steg-analysis. The technique used to decode the message of cryptography is reverse engineering.

2 Related Works

Author in [1] proposed to shred the main video into multiple frames. A single discrete level transformation is done to select single frame and the secret image file. A private key is used to maintain security. Arnold fiction is used to embed and obtain data. Then, the inverse discrete wavelet transformation is used to obtain the reverse steganographic file. Various parameters like PSNR and MSE are obtained to judge the quality of the video.

Author in [2] proposed to apply 2D-DCT algorithm to the main video and encode the data. The PSNR value is obtained to check the quality of the video post manipulation. The information is embedded according to the DCT coefficients obtained by the method.

Author in [3] uses the least significant bit technique and video file as a carrier to carry a secret message. An extra layer of security is provided to the process by generating a random index generating sequence and the data in hidden according to this sequence. The original and manipulated images cannot be differentiated by the naked eye.

Author in [4] has used advanced technique for encryption of images using discrete Haar wavelet form. HDWT form provides fewer disturbances and fewer changes and also decreases the complexity of encoding. In this process, only one-fourth of the cover image contains the secret message, and the other three regions carry less important details of the message.

Author in [5] has used 2D-DCT method and BCH codes to encrypt data into video file. Embedding is done using the DCT coefficients of the frames. Y, U and V planes are used to encode data, and the DC coefficients are ignored. An algorithm is tested for both sow and fast-moving video types. The hidden ratio is found to be around 27.53%, for effective hiding capacity and lessen the video quality distortion. The effectiveness is tested under various penetration tests.

Author in [6] proposed the method of video steganography and its use in insecure communication medium. Random byte hiding and LSB methods are used in this process. In random byte hiding method, lines are hidden in different places in the video frames. In the LSB technique, the secret message is hidden in the least significant bit of every frame.

3 Proposed Work

3.1 Proposed Work for Image Steganography

3.1.1 Image Steganography

Image steganography is the process of hiding a secret data or image or message into a cover object that is an image file. In image steganography, the cover image is not altered at all and resembles the original cover image when viewed through naked eyes.

Image steganography has a wide variety of application starting from digital marketing to secured data transmission and in numerous fields. Even defense and armed forces can make very efficient use of steganography to make there highly confidential data and information encoded with Steganography techniques and transmit it over countries, state and territories. Thus, image steganography is no doubt a very useful process of data hiding.

3.1.2 Simple Least Significant Bit (LSB) Method

In this method, the user provides us with the original image file and the secret text message to be encoded. In this model, we convert the message into its corresponding binary format including all spaces and special characters. Now, using open in this method, the user provides us with the original image file and the secret text message to be encoded. In this model, we convert the message into its corresponding binary format including all spaces and special characters. Now, using OpenCV and NumPy module, we extract pixel values; we take three pixels at a time and encode it with a single character of the binary string (suppose 10100111). Here, we check the value of the image pixel corresponding to 0 or 1 is even or odd. Let say that the third bit in the binary format is 1 and the pixel value is an even number, then the new pixel value is pixel = pixel + 1.

Else if it is odd, then the pixel value remains the same. Similarly, if the nth position bit is 0 and the nth pixel value is odd, then pixel = pixel + 1.

Else if it is even, then the pixel value remains the same.

Likewise, we continue the process until the last character of the message gets embedded into the image file, i.e., the pixel values are changed according to the letters binary format. And we finally save the cover image in the given location.

3.1.3 Algorithm for the Proposed Method

The algorithm for the proposed method is in the following steps;

1. Select message to be encoded.
2. Select the image file.
3. Check the size of the encoded and original images.
4. If original image > 7 times of the secret message.
5. Convert the individual letters of the message to binary values inclusive spaces.
6. If encoded data size > original image size.
7. Select the original image such that size should be large and go to fifth step.
8. Now, alter the least significant bit of the pixel values in order of 0/1 of the message.

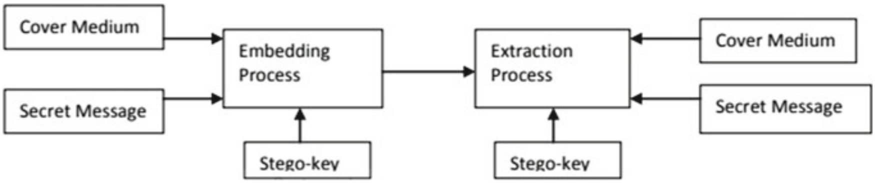


Fig. 1 Block diagram for video steganography

9. Check the intensity of pixel. If value of the pixel >3, embed three LSB in original pixel, else embed three LSB in original image file.
10. Save the encoded image file.

3.2 Proposed Work for Video Steganography

Here in this article, we will discuss some of the methods or techniques. In the modern world, secured data transmission is a matter of concern for all is it banks, offices or government offices due to the development of the Internet and multimedia contents such as audio, image and video. Transmitting secret images, secured data should be taken into concern because if by any chance any hacker bypasses your security and takea out your valuable information. Video steganography is the process of hiding confidential data into a cover object that is a video film and without making any visible difference into the video file that is any change of pixel of video frames must be negligible as shown in the flowchart in Fig. 1. In this paper, a review on various video steganography techniques has been presented.

3.2.1 Data Inclusion Using LSB Method

In this model, the user provides us with the secret text message and the video film (cover file), then we take separate individual image frames of videos, calculate the frames per second(fps) of the video file and also determine the codec of the video file (the degree of compression applied). Next, we convert the entire text message into its corresponding binary format. Now, we start the encoding process where we take three pixels at a time starting with the first pixel and encode it with a single letter of the text in bits (suppose 10110110). Here, we check the value of the image pixel corresponding to 0 or 1 is even or odd. Let say that the third bit in the binary format is 1 and the pixel value is an even number, then the new pixel value is [pixel = pixel+1] else if it is odd, then the pixels value remains the same. Similarly, if the nth position bit is 0 and the nth pixel value is odd, then [pixel = pixel + 1] else if it is even, then the pixel value remains the same.

The result of a sample three-pixel value originally

00001101 00011101 11111001
10000110 00011111 11011010
10001111 00110000 11011011

will result in

00001101 00011100 11111001
10000111 00011110 11011011
10001111 00110000 11011011

likewise, the entire message is encoded, the pixel values are changed according to the message binary form till the end. After the completion of this process, we again save the individual encoded video frames at a given location and rejoin the video frames with the same fps rate and with HFYU codec so that the video appears to be entirely same as the original video file, and it gets saved at the user-defined location.

Now, to decode the given video file, we use reverse engineering, we extract the video frames, and then, we take three pixels at a time and check whether it is odd or even, we take another variable of string type to store the binary format of the decoded message. If the value of the pixel is odd, we put [bin = bin + 1] else if it is odd pixel value, then we add 0 [bin = bin + 0], we finally typecast bin to int, and we find its corresponding ASCII value and store it in another string type variable message. And we continue this process until we extract the message from all the encoded pixel value, and finally, we display it.

3.2.2 Flowchart for the Proposed Method

See Figs. 2 and 3.

4 Result and Analysis

In this project, we are using steganography method to secure data inside image and video file as explained in proposed work. First of all, for an image, we hide the text inside the pixels of the image, in which, we observe no change in the photo after encoding data in it as shown in the below example in Fig. 4. This can store as much as a page of data in a single image file without any visible change in it. We are applying the above method in the video also. So, for the video, we can store large text documents inside it which will make data more secure and safe. This can be used to hide high confidential data so that no one can think of retrieving it from a video. This technique will only be used by one person to hide it and send file anywhere, and he

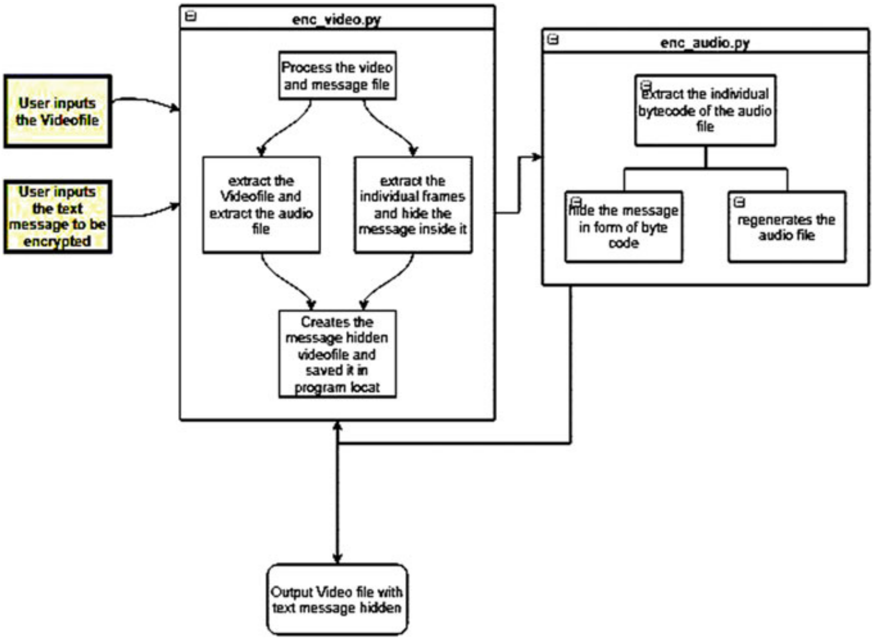


Fig. 2 Block diagram of video steganography for encode function

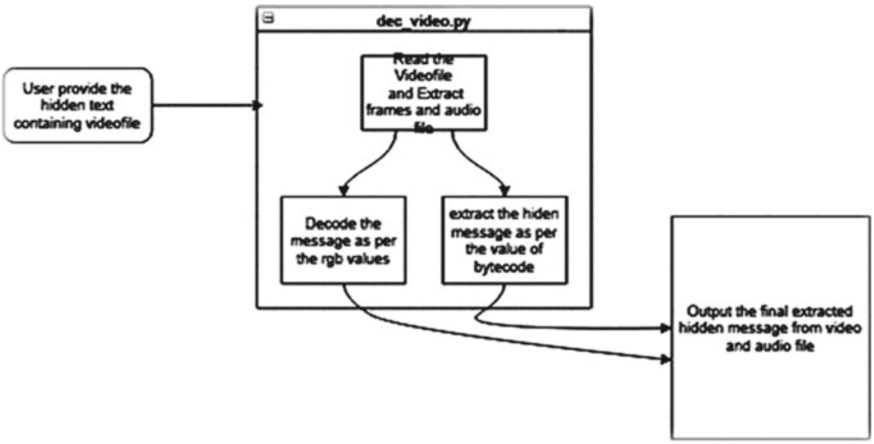


Fig. 3 Block diagram of video steganography for decode function

can access (like public pc/Internet parlor) it from there without being noticed by any other person who else uses that pc or system at it leaves no traces of it while decoding the message. Since we cannot attach output file for video, it is working flawlessly, i.e., no visible change in the video. But the only problem we are facing in encoding

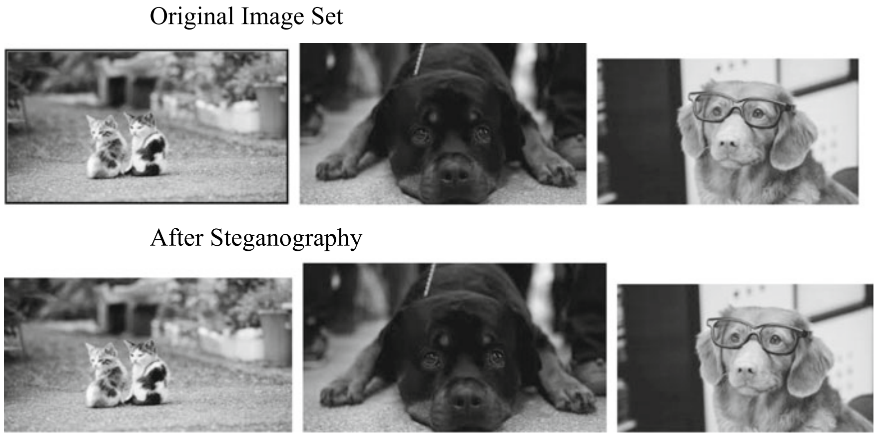


Fig. 4 No observable changes after stagnating the original image set

the image and video is that the size (here size refers to the memory captured by the file) of image is comparatively minimally increased as of actual size, and for video, its size increases by large amount as compared to its actual size (greater than 200%). The above problem is due to the use of HFYU codec which does not compress the file and data inside it, in future, we are trying to solve this problem.

1. Output for image encoding

5 Conclusion and Future Work

Through this paperwork, we have come across an efficient and reliable method for image and video steganography which can be broadly used in different fields like private communication and information storage, preventing data manipulation, digital content distribution management, media database system for and numerous fields of applications. Besides the application of steganography, video steganography can be efficiently used by the defense and armed forces for secured data transmission and data storage of their highly confidential data and information. This paper also gives us a brief introduction to digital image processing through Python module OpenCV and pixel significance and pixel alteration algorithm.

After this paperwork, we look forward toward text in audio steganography, an image in image steganography, video in video steganography, text in text steganography and also building a more secure and efficient algorithm to implement steganography. We also are working to fix the issue of the enlarged video file (nearly 200% of its original memory size) after implementing encoding the message onto it.

References

1. Thakur, A., Singh, H., Sharda, S.: Secure video steganography based on discrete wavelet transform and Arnold transform. Int. J. Comput. Appl. **123**(11), 25–29 (2015)
2. Rajesh, G.R.: Hiding data in images by Simple LSB substitution. J. Pattern Recogn. Soc. **37**(3), 469–474 (2013)
3. Rajesh, G.R, Sajin Narguman, A.: Steganography algorithm based on discrete cosine transform for data embedding into raw video streams. In: Chennai Fourth International Conference on Sustainable Energy and Intelligent System 2013
4. Houssein, E. H., Ali, M.A.S., Hassanien, A.E.: An image steganography algorithm using Haar discrete wavelet transform with advanced encryption algorithm. In: Federated Conference on Computer Science and Information System (2016)
5. Mastafa, R.J., Elleithy, K.M.: A DCT-based robust video steganography method using BCH error correcting codes. IEEE (2016)
6. Bhole, A.T., Acharya, A.K., Yadav, V.K., Batham, S.: Hiding large amount of data using a new approach of video steganography. In: 4th International Conference on the Next Generation Information Technology Summit (2013)

Intelligent Intrusion Detection Using TensorFlow

Adarsh Padhi, Gyanaranjan Sahoo, and Satyabrata Maity

Abstract Detection of human beings accurately in a visual system like in images, videos, live stream videos is critical for diverse application which includes elderly abnormality detection, violent detection, human gait characterization, congestion analysis, person recognition and gender classification. Object detection is the process of distinguishing the presence of predefined types of objects in a frame. This process includes both identification of the presence of the objects with a rectangular bounding surrounding it. The application utilizes MSCOCO model trained from ImageNet database with TensorFlow to detect human beings or objects with a greater accuracy. For greater speed in detection, single shot detectors (SSD) have been used for detection. There is a tradeoff between faster detection and accuracy. The test observations results show the detection as well as the recognition of the human beings with a greater speed and accuracy that indicates the effectiveness of the proposed approach.

1 Introduction

Most of us in our daily life are worried about the thieves, accidents, murders, etc. Everyone is aware of this and has love for their life. So, they appoint Watchmen, Security Guards, Bodyguards in order to protect their lives'. Nowadays, people were much serious with their belonging. So everywhere, a CCTV camera is being installed in order to capture all the things that is a beyond human capability.

Let us assume,

A. Padhi (✉) · G. Sahoo · S. Maity
Computer Science & Engineering, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India
e-mail: adarshpadhi14@gmail.com

G. Sahoo
e-mail: gyanaranjan7205@gmail.com

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_66

1. There is a robbery, and the user does not have sufficient time and patience to review it.
2. There is less storage space available for storing CCTV videos.
3. There is fire in an unattended house or is about to have any lethal accident due to weapons.

We have overcome above the situation, by object detection [1], it could really helpful and can bring a change in our society. We have designed a software that can possibly overcome the problems mentioned above.

Machine learning [2] is a field of computer software study in which the user trains the computer or simply we can say that it makes the computer to think as well as learn. It is a part of artificial intelligence that provides systems the ability to learn automatically and enhance from past without going deep into program. Machine learning throws light on the development of computer programs which can have access data and manipulate it in learning for themselves. Machine learning algorithms are often distinguishing as supervised or unsupervised. Object recognizing is a specific term to elaborate a collection of specific computer vision works which include identifying objects in virtual scenario. Image categorization involves predicting the class of one object in the frame utilizing TensorFlow [3] in this case.

Motion detection is a technique of detecting a changed frame of an object with respect to its environment or a change in the environment with respect to an object. It also implements face detection which is a software used in a various of applications in order to identify human face in digitally like in images, videos, etc. We can also define as the skeletal process in which humans are located and attend their faces in a virtual scene. In this paper, we will be explaining the basics and applicability of human detection and the effectiveness of this technology for taking valid frame footage, sharing of data through messages and online streaming which were required for the user to be alert and can take precaution before dealing with the situation.

2 Related Work

The author in [4] described a number of ways utilized in object detection such as background subtraction which detects object by difference in current and reference frame in block by block fashion, optical flow approach which predicts motion in video by comparing frames with other frames, temporal analytics of gait motion.

The authors in [5] used background subtraction methods to detect human and track in different levels of illumination. The foreground is captured by subtracting previous frames from a test frame by checking pixel values.

The authors in [6] introduce a method for human detection and action recognition which combines Euclidean distance and joint entropy-based features selection and uniform segmentation. It segments the moving objects in a frame by fusing novel uniform segmentation and expectation maximization, and then, it extracts a new set of fused features using local binary patterns with HOG and Haar-like features.

The features are selected by novel Euclidean distance and joint entropy-PCA-based method and are classified using a multi-class support vector machine.

The authors in [7] utilize Haar boosting-based methods for moving human detection. He named the method as Haar boosting--HOG--SVM detection process to detect pedestrians. In this method, Haar-like characteristics eradicate frame, and then, it is distinguished with the help of boosting-based method.

3 Proposed Solution

We have designed a software which will reduce the time as well as the data consumption rate. The project "Intelligent Intrusion Detection" perfectly defines itself. In this project, we have used machine learning which enables to detect between humans and all other species and objects. We have used TensorFlow with MSCOCO [8] model.

TensorFlow is an end-to-end open source software library which is used to create the dataflow across a range of tasks. It has libraries performing advanced mathematical functions. It is used for machine learning applications especially in neural networks. This detection concept is basically based on the neural networks basically RCNN [9] i.e., (region-based convolution neural networks). It develops boxes and distinguishes every frame for each instance of object with respect to the image. It follows feature pyramid network [10] (FPN) algorithm and a ResNet101 backbone.

The application takes a video or webcam stream as an input and writes out the frames containing any human intrusion, thereby making a brief video. This result in shortening of original video and data consumption rate, which leads to save on time and storage space.

This project also plays a vital role in protection, due to its double layered security. Firstly, when it finds the intrusion, the first level of security gets activated, and it sent a message or a call to the owner about informing that there is some suspicious in your house. If the application detects weapons like (knife, guns, fire), then the second level of security gets activated, and it sends a message to owner and on confirmation or no response from owner, for over 10 min, it sends location and an alert message to the nearby police station.

The application also supports live streaming of webcam or video input over Internet to platforms like YouTube and Twitch.

4 Prototype

The vital step in human detection of a smart surveillance system is distinguishing among moving objects in a frame. The successful interpretations of higher level human motions greatly rely on the precision of human detection. The detection process occurs in two steps: object detection and object classification by deep learning techniques. Among the deep learning-based techniques, two broad classes of methods

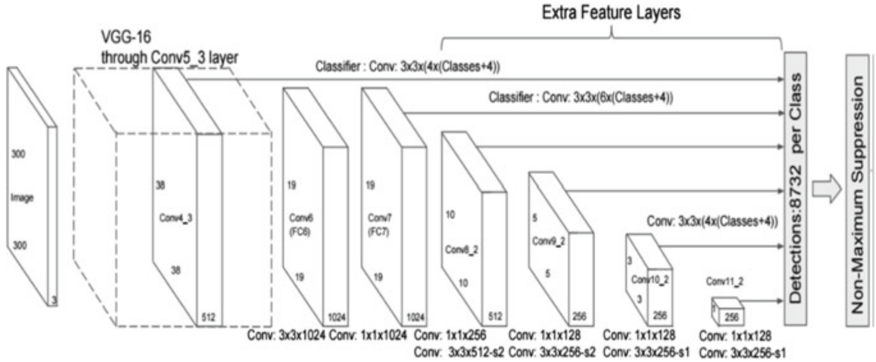


Fig. 1 SSD architecture for the proposed approach [11]

are prevalent: two stage detection (RCNN, Fast RCNN, Faster RCNN) and unified detection (Yolo, SSD).

Single shot detector (SSD) [7] model has been used in this project. The SSD model does two tasks—extracting feature maps and applying convolutional filters to detect object. There are six auxiliary convolutional layers which are used to detect object of various sizes in image. The SSD model makes about 8732 predictions in total using those layers. The architecture is shown below in Fig. 1.

The SSD converts the visual geometry group (VGG) model into a fully convolutional neural network. We further attach some convolutional network to handle bigger objects. The VGG network outputs a 38 × 38 feature map. Then, the added layers produce feature maps of 19 × 19, 10 × 10, 5 × 5, 3 × 3, 1 × 1. For the prediction of bounding boxes of various sizes, all of these feature maps are used. The later filters could be used for larger objects. Several activations have been passed to the sub-network that acts as a classifier and a localizer.

Anchors are a collection of predefined box shapes overlaid on image usually with different scales and aspect ratios. They act as reference points on ground truth images. A model is trained to make two predictions for each anchor---a distinct class to which it belongs, a continuous offset by which the anchor needs to be shifted to fit the ground truth bounding box.

The SSD tries to match the ground truth annotations with anchors during the training period. A number of anchors are associated with each element of future app. Then, the IoU, otherwise called Jaccard distance, which is the ratio of area of overlap of anchor and ground truth annotations to its area of union. A match is considered only if the IoU value is more than 0.5. The loss functions that have been used here are multi-box classification and regression loss. The softmax cross entropy is used for the multi-box classification loss. Likewise, the smooth loss is used for regression loss. The base network of the SSD model is derived from RESNet and it consists of modified convolutional layers for fine-tuning and then the classifier and localizer networks. The resultant deep network is trained end-to-end on the dataset.

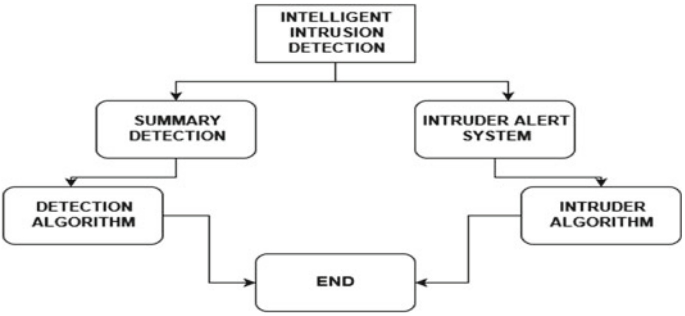


Fig. 2 Structure of project for intelligent intrusion detection

The project can be used in two different modes exclusive to each other, i.e., storage mode, surveillance mode. Figure 2 shows workflow. The storage mode is implemented by using summary detection and surveillance mode by intruder alert system.

Storage Mode: This mode writes only those frames in which any human intrusion is detected to a video file along with current timestamp. The video source can either be a video file or a live webcam. This will help user in several ways like—producing a brief output video such that it can be reviewed fast thereby saving time, taking a much smaller storage space than the original video should have taken thereby saving resources and storage space. The workflow can be seen in Fig. 3. It starts by breaking video to frames and loops over each frame to find out any human or objects. The parts of video identified as containing human activity are written to a video file.

Surveillance Mode: Under this mode, the application actively takes part in security of user as shown in Fig. 4. It has two levels of security. In first level, it informs the user when it finds any human locomotion in its area of vision by either a SMS or a call. On second level, when found any weapons like knife, daggers, guns or fire, it informs the user by a message with a link to deactivate the alarm, if the alarm is not deactivated in 10 min, the location along with an alert message is sent to nearest police station. SMS/call is sent by utilizing API provided by Gateway. Live stream of video to Web sites like YouTube and Twitch was also made possible by using ffmpeg library.

A short video of a burglary on road was taken from YouTube. From the video, 150 frames were taken out in total at regular interval. Four such tests were done, and results were manually verified. The results obtained were classified into four parts.

Total Positive: When the experimental results show that a frame **contains** a human and in actual data there **is** a human, it is considered to be a Total Positive case.

False Positive: When the experimental results show that a frame **contains** a human and in actual data there **is no** human, it is considered to be a False Positive case.

False Negative: When the experimental results show that a frame **does not contain** a human and in actual data there **is** a human, it is considered to be a False Positive case.

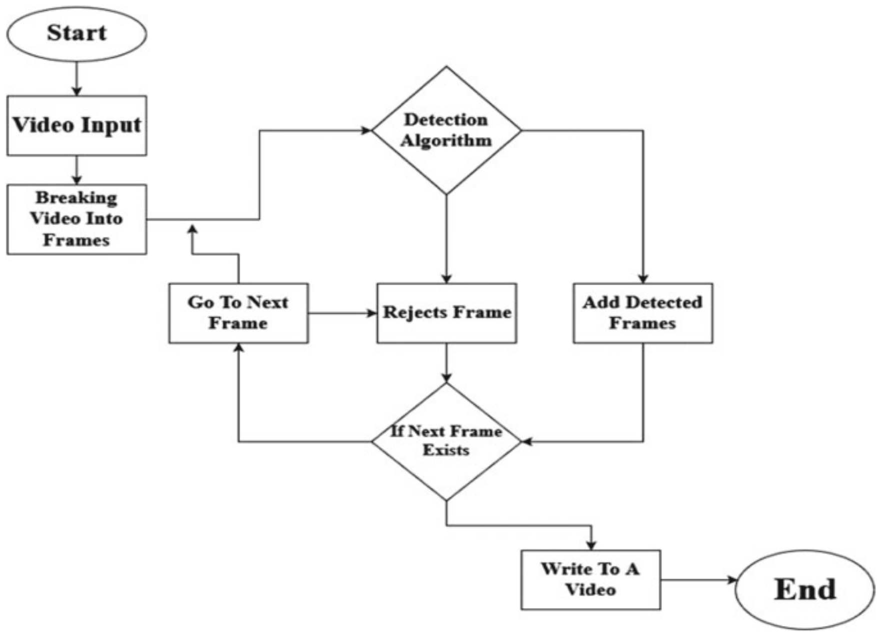


Fig. 3 Workflow of storage mode to pursue the summary of the video

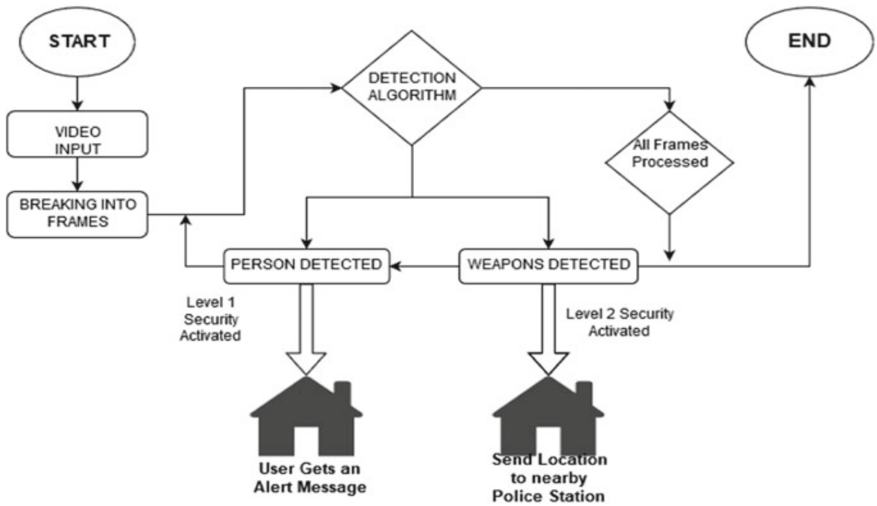


Fig. 4 Workflow of surveillance mode

Total Negative: When the experimental results show that a frame **does not contain** a human and in actual data there **is no** human, it is considered to be a False Positive case. The graph includes recall versus precision is shown in Fig. 5. It shows how well the human detection has taken place by the proposed approach. On the other hand, Fig. 6 shows the graph containing specificity versus precision. Table 1 shows the confusion matrix for the various video sample taken from ImageNet dataset. It describes the accuracy of the trained model with respect to the predicted cases over Actual cases. The accuracy was found to be 0.81. Table 2 shows the performance matrix which is the comparison between specificity, precision, and recall for the various video sample taken from ImageNet dataset. The average specificity, precision and recall are 0.973025, 0.984075, 0.7478 respectively.

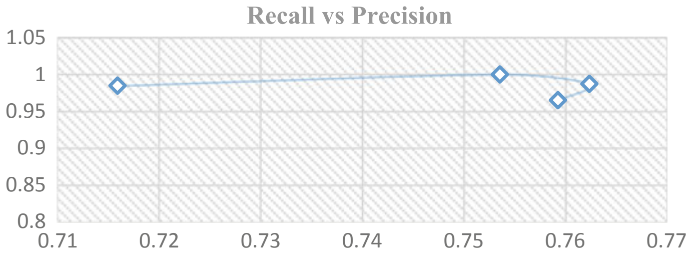


Fig. 5 Recall versus precision

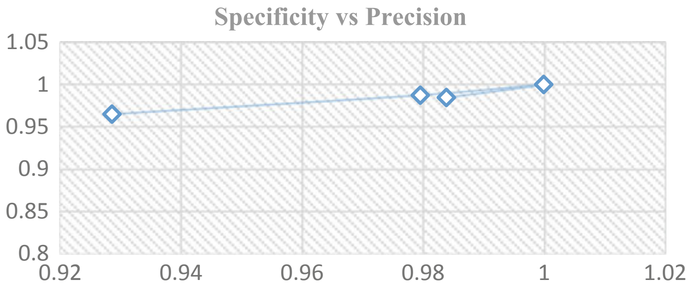


Fig. 6 Specificity versus precision

Table 1 Statistical analysis of the proposed approach using ImageNet database

Test count	True positive	False positive	False negative	True negative
Video segment I	82	3	26	39
Video segment II	77	1	24	48
Video segment III	104	0	34	12
Video segment IV	63	1	25	61

Table 2 Parametric analysis of the proposed approach with ImageNet database

Test count	Specificity TN/(TN + FP)	Precision TP/(TP + FP)	Recall TP/(TP + FN)
Video segment I	0.9286	0.9647	0.7593
Video segment II	0.9796	0.9872	0.7624
Video segment III	1	1	0.7536
Video segment IV	0.9839	0.9844	0.7159

The project can be used for several purposes and has several use cases. Some of them are

It can be used in active or passive mode for saving storage space by saving only those frames when human is detected and live stream them. It can be used in research purposes (in places where you wait for an event to happen) to inform you when that event happening like monitoring tigers near forest areas, etc. The application can work intelligently to detect any causalities like fire, accidents, robbery and put on alarm or inform the owner also the police station so that necessary actions can be taken.

A few glances of detection from application are shown in Fig. 7a, b.

Fig. 7 **a** Shows a person detected in camera with 54% and **b** probability and 87% probability, respectively

5 Conclusion

A convincing amount of work has been completed with a view to detect people in a CCTV footage. However, the low-resolution frames from the footage make this application to perform better. With the machine vision reference, it is difficult to recognize an object as a human because of its number of possible outcomes. But a combination of several features could be useful identifying human. The application can be further improved to work with multiple cameras and process frames of each time frame from different angles to provide better results, accuracy and lesser false negatives. Although the application is not 100% accurate still, it has a better accuracy in comparison with speed and memory usage. This application could be a great help to people along with push the current real-world implementation techniques a bit nearer to the world of automation.

The project can be further improved in various ways and will be considered as future work. Some of them are---first, detecting abnormal or violent activities like multiple-person interactions (e.g., fighting and personal attacks), accidents on the roads, elderly people abnormal monitoring. Second, person detection in dense crowds and people counting and pedestrian detection which could be helpful in a variety of other application. Third, tracking a specific person in a CCTV footage by utilizing using face recognition and gait recognition techniques. Fourth, automatic detection of a fall for elderly people is one of the major applications which can be implemented to human detection in surveillance videos. Fifth, a live stream platform in which a user can have access to watch the video at any place of the world with and an Internet-enabled device.

References

1. Amit, Y., Felzenszwalb, P.: Object detection. In: Ikeuchi, K. (ed.) Computer vision. Springer, Boston, MA (2014)
2. Liu, L., Ouyang, W., Wang, X., et al.: Deep learning for generic object detection: a survey. Int J Comput Vis (2019). https://doi.org/10.1007/s11263-019-01247-4
3. Toleubay, Y., James, A.P.: Getting started with tensorFlow deep learning. In: James, A. (eds.) Deep Learning Classifiers with Memristive Networks. Modeling and Optimization in Science and Technologies, vol. 14. Springer, Cham (2020)
4. Paul, M., Haque, S.M.E., Chakraborty, S.: EURASIP J. Adv. Signal Process. **2013**, 176 (2013). https://doi.org/10.1186/1687-6180-2013-176
5. Sardeshmukh, M.M., Kolte, M., Joshi, V.: Performance analysis of human detection and tracking system in changing illumination. In: Corchado Rodriguez, J., Mitra, S., Thampi, S., El-Alfy, E,S. (eds,) Intelligent Systems Technologies and Applications 2016. ISTA 2016. Advances in Intelligent Systems and Computing, vol. 530. Springer, Cham (2016)
6. Sharif, M., Khan, M.A., Akram, T., et al.: J Image Video Proc. **2017**, 89 (2017). https://doi.org/10.1186/s13640-017-0236-8
7. Oliver, J., Albiol, A., Morillas, S., Peris-Fajarnés, G.: A real-time person detection method for moving cameras. In: Corchado, E., Wu, X., Oja, E., Herrero, Á., Baruque, B. (eds,) Hybrid Artificial Intelligence Systems. HAIS 2009. Lecture Notes in Computer Science, vol. 5572. Springer, Berlin, Heidelberg (2009)

8. Lin, T.Y., et al.: Microsoft COCO: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision—ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol. 8693. Springer, Cham (2014)
9. Yamashita, R., Nishio, M., Do, R.K.G., et al.: Insights Imaging **9**, 611 (2018). https://doi.org/10.1007/s13244-018-0639-9
10. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision—ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol. 8691. Springer, Cham (2014)
11. https://www.researchgate.net/figure/The-model-of-Single-Shot-MultiBox-Detector-SSD-25_fig5_332948824

Omniscient Eye Student Attendance Using Facial Recognition and Classroom Behavior Analysis

Debashish Mishra, Omm Mishra, and Satyabrata Maity

Abstract The primary aim of this application is to be able to record the attendance of students inside a classroom without having to roll call each student individually and upload the attendance data to the cloud so that a backup is maintained at all times and is easily accessible from any device with an Internet connection and required permissions. This application also aims to reduce the load of taking attendance manually inside a class and helps the teacher record the behavior of the students to infer how effective the lecture is. The application provides the teacher with an insight into his/her own teaching patterns and habits which will help the teacher to rectify any issues that may be a cause of the student's disinterest in the subject.

Keywords Facial recognition · Cloud data · RCNN · Masking · Pixel segmentation · Firebase · Image segmentation · Biometric data · ResNet

1 Introduction

The conventional method of recording attendance in any institute nowadays is to call out the name or roll number of the students of the particular subject. However, recording attendance this way leads to a huge wastage of precious lecture time. This becomes a major issue for classes with high number of students. Attendance management also becomes a tedious task. There is also the issue of students marking proxy attendances for their peers.

D. Mishra (✉) · O. Mishra · S. Maity
Department of Computer Science and Engineering, Department of Computer Science and Information Technology, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha 751030, India
e-mail: d3bashishm1shra@gmail.com

O. Mishra
e-mail: ommmishra830@gmail.com

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_67

Automatic attendance marking can solve issues such as the errors caused when attendance is taken manually, or when the attendance records are fed to the database after reading from the sheet. Automatic attendance also helps the teacher catch proxy attendances, hence ensuring discipline in the class.

The teachers nowadays are unable to determine exactly how many students do not have any interest in the lecture or how boring the lecture itself is. They try to brute force their way out of the lecture and sometimes skip some parts of the lecture when they see more than half of the class seem to be uninterested in the lecture.

A student's interest in a subject plays a huge role in his education and determines if he/she will be able to actually take something useful with him/her after the class ends and implement it in his/her own projects.

Therefore, to summarize, the application helps us to avoid wastage of valuable lecture time inside a class that usually gets spent on taking attendance and evaluating if there are any fallacies in the roll calls and to provide the teacher with a summary of how many students are interested in the subject lecture which will help the teacher to improve the quality of the lecture if need be which in turn will greatly improve the performance of the students in the examinations of their respective subjects.

2 Related Work

In the paper, An Efficient Automatic Attendance System Using Fingerprint Reconstruction Technique [1], the authors mention that "Biometrical attendance system is one of the most successful applications of biometric technology," which is true as biometric systems help us to avoid proxy attendances thanks to the uniqueness of fingerprints. Fingerprint recognition is a well known and established field today, but still identifying an individual using his fingerprint by scanning through all the fingerprints in the database is a time-taking process. Moreover, fingerprint scanners never successfully scan a fingerprint on the first try. Therefore, we had to try and search for a better alternative which will consume less time and will also record the emotions and behavior of the students.

In the paper, Unstructured Human Activity Detection from RGBD Image [2], the authors perform detection and recognition of unstructured human activity in unstructured environments. They use a Red-Green-Blue-Depth (RGBD) sensor as the input sensor and use human poses and actions as features for their model.

This paper inspires us and showed us a path that we can take to achieve our goals; however, the limitation here is that not every CCTV camera is equipped with a RGBD sensor which makes it difficult to implement inside any university.

Yolact [3] is a fully convolutional model for real-time instance segmentation. It has been trained on the MS COCO dataset just like Matterport's Mask-RCNN model [4]. Although it has 29.8 mAP as compared to Mask-RCNN's 38.3 mAP, the inference is very fast in Yolact. Mask-RCNN is tested for 6.0 FPS whereas Yolact has real-time instance segmentation capabilities with test results as 33.5 FPS.

As compared to Mask-RCNN, we found out that the masks produced by the Yolact model were not very refined. The current implementation by the authors only supports GPU inference and training.

Dlib [5] is a toolkit which can be used to make machine learning and data analysis applications for the real world, in C++. The most popular application of deep metric learning is face recognition. It is a ResNet network with 29 convolutional neural layers and is similar to ResNet-34 network from the paper Deep Residual Learning for Image Recognition by He et al. [6] with an exception of a few layers and half the number of filters per layer.

This tool is used by Geitgey for his Python library *face_recognition* [7] which has helped us make facial recognition much easier.

3 Proposed Work

The dataset provided to the models for training and testing was salvaged by taking pictures of classes in progress and laboratories as described in Fig. 1.

A brief explanation of the steps involved:

1. First, snaps of the classroom after every 10 min are analyzed and used for facial recognition. The faces recognized in the majority of the images are provided with the attendance. If someone were to attend only the first half of the class and then ditch the rest, then he will not receive any attendance. The attendance is then fed to the database hosted on the cloud.
2. The next step is to predict or guess the behavior of the students in the class by detecting the emotions on the faces of the students and analyzing the skeletal poses of the students. This analysis helps us to review the lecture of the subject being taught and provide the lecturer with a result of how many students were interested in the class and an average interactivity factor of the classroom.
3. It is made sure that the data predicted by the model at any point of the process is kept anonymous. If the teacher asks for a disclosure, then he can be provided with the data after taking permission from the concerned authorities.

3.1 Masking Over Humans

The model being used in our applications is pre-trained on the MS COCO dataset which contains the images, along with the bounding box coordinates and the mask positions. The images are then fed to the model, which outputs a Python list containing a dictionary. This dictionary contains the regions of interests (RoIs) and the classes of different entities detected along with their scores. The dictionary also contains masks for different entities created.

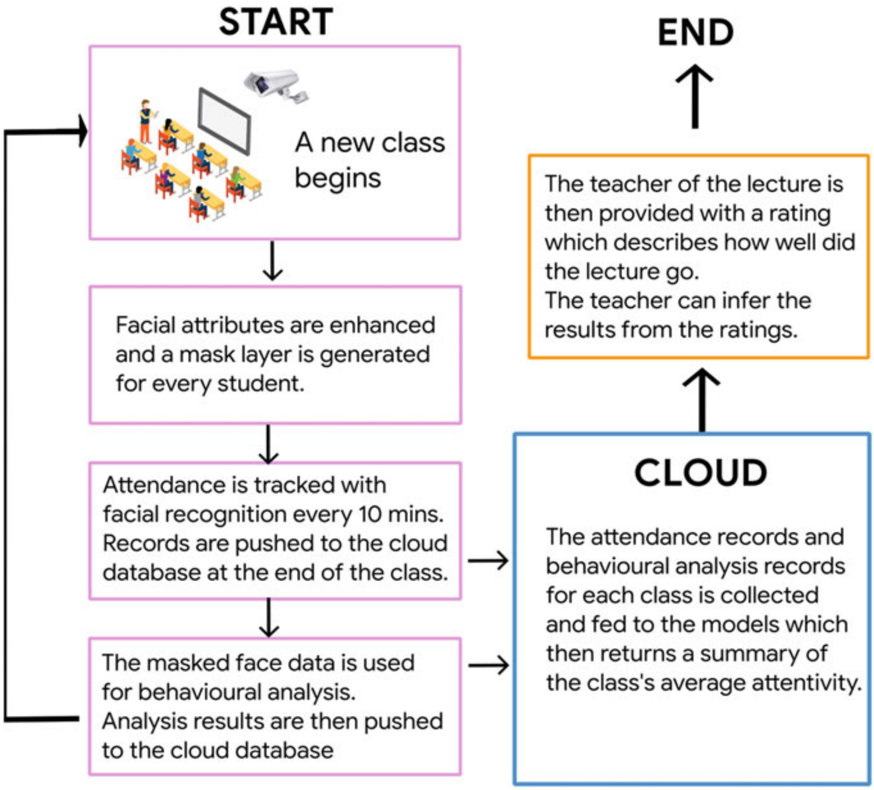


Fig. 1 A workflow diagram briefly describing how the application works

The masks are in the form of arrays containing Boolean values. We take these values and apply it on the original image to separate each entity from the rest of the image to run facial recognition and detect emotions. This is done by taking the mask values of a specific entity and mapping them to the original image.

The pre-trained model used here has been provided by the Matterport organization [8]. The model is based on the paper Mask RCNN for object detection and segmentation [4] as shown in Fig. 2.

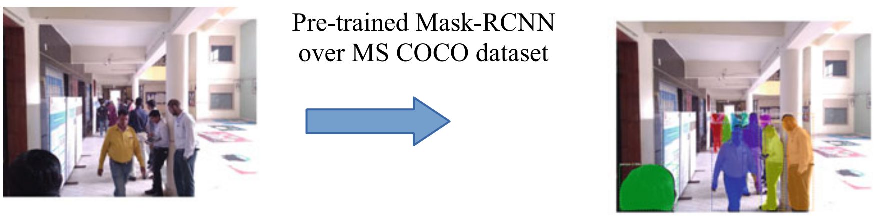


Fig. 2 Masking over individuals using the pre-retrained mask-RCNN model

The pre-trained model has a function *visualize.display_instances()* which requires the following parameters:

1. The source image
2. An array of region of interests.
3. An array of masks.
4. An array of the IDs for different classes in the image.
5. The class names of the classes whose IDs have been provided.
6. An array for prediction scores.

Function 3.1: *Matterport's function to display instances on the image provided.*

```
1    visualize.display_instances(image, r['rois'], r['masks'], r['class_ids'],
                                 class_names, r['scores'])
```

When we provided the above function with the input image we receive this output.

OUTPUT:

```
{'rois': array([[1662, 1562, 3095, 2369],
       [1512, 2713, 3094, 3218],
       [1510, 1536, 2159, 1737],
       [1475, 2063, 2831, 2428],
       [1645, 2328, 3085, 2741],
       [2344,   54, 3094,  994],
       [1521, 1708, 1983, 1881],
       [1492, 2021, 1748, 2162],
       [1496, 1869, 1794, 2001],
       [1525, 2344, 1736, 2528],
       [1502, 1677, 1680, 1767]], dtype=int32), 'class_ids': array([1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1], dtype=int32), 'scores': array([0.99956316, 0.9995369 , 0.9993724 , 0.99840987,
0.9978783 ,
       0.9962592 , 0.98906416, 0.97148347, 0.9624833 , 0.85435385,
       0.7886227 ], dtype=float32), 'masks': array([[[[False, False, False, ..., False, False,
False],
        [False, False, False, ..., False, False, False],
        [False, False, False, ..., False, False, False],
        ...,
        [False, False, False, ..., False, False, False],
        [False, False, False, ..., False, False, False],
        [False, False, False, ..., False, False, False]],
        ...,
        [False, False, False, ..., False, False, False],
        [False, False, False, ..., False, False, False],
        [False, False, False, ..., False, False, False]]])}
```

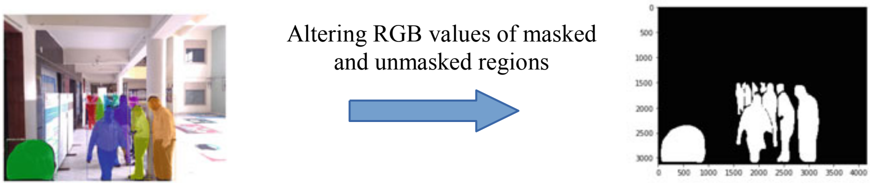


Fig. 3 Reducing the pixel RGB values of the unmasked region to (0, 0, 0) and increasing the pixel RGB values of all masked region to (255, 255, 255)

3.2 Reducing Background Noise

Now that we have the individual humans highlighted, we can now try to reduce the background noise by reducing the pixel RGB values of the unmasked region to (0, 0, 0) and increasing the pixel RGB values of all masked region to (255, 255, 255). This creates an illusion of separation between the background and the humans in frame as shown in Fig. 3.

Function 3.2: *Function for extracting only B/W outlined image.*

```
def apply_mask(image,masks):
    image.fill(0)
    for i in person_detections:
        mask = masks[:,:,i]
        image[:,:,0] = np.where(mask==1, (image[:,:,0] * 0) + 255,
                                image[:,:,0])
        image[:,:,1] = np.where(mask==1, (image[:,:,1] * 0) + 255,
                                image[:,:,1])
        image[:,:,2] = np.where(mask==1, (image[:,:,2] * 0) + 255,
                                image[:,:,2])
    return image
```

3.3 Masking Over a Single Entity

The above solutions seem to have solved our problem; however, we do not want the masks over the humans to be completely white. A simple solution for this would be to not alter the RGB values of the masked region; however, that does not help us to separate a single individual from the rest of the crowd. Therefore, instead of increasing the RGB values of all the masked regions to (255, 255, 255), we decrease the RGB values of all the masked region except the mask we are concerned with to (0, 0, 0) as shown in Fig. 4.

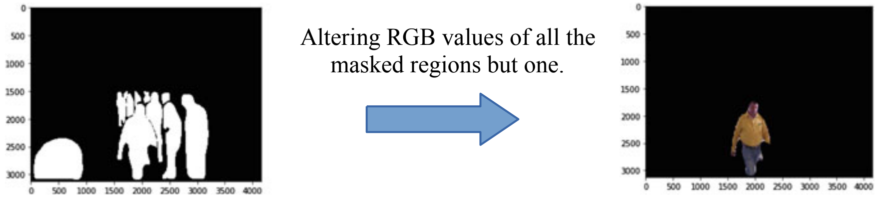


Fig. 4 Reducing the pixel RGB values of the unmasked region and the masked region to (0, 0, 0) except for the masked region we are concerned with

Function 3.3: *Function for extracting single entity.*

```
def one_entity(aImage, masks):
        mask = masks[:,:,0]
        aImage[:,:,0] = np.where(mask==1, aImage[:,:,0] * 1, aImage[:,:,0] * 0)
        aImage[:,:,1] = np.where(mask==1, aImage[:,:,1] * 1, aImage[:,:,1] * 0)
        aImage[:,:,2] = np.where(mask==1, aImage[:,:,2] * 1, aImage[:,:,2] * 0)
        return aImage
```

3.4 Implementing Facial Recognition on a Single Entity

Facial recognition is then implemented on this noise-reduced image to get the most accurate results. However, as you can see, the resolution of the image decreases with each step because of which the facial recognition algorithm is unable to successfully detect any faces as shown in Fig. 5.

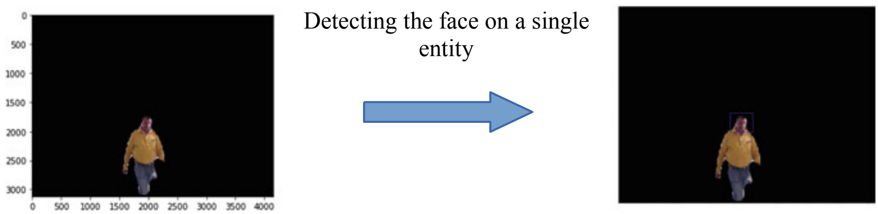


Fig. 5 Detecting a face on a single entity image. We used a modified version of the example code provided by Geitgey on the face_recognition repository

Function 3.4: *Function for detecting a face in the image.*

```
image = face_recognition.load_image_file("iamge location")
image_facial_encoding = face_recognition.face_encodings(image)[0]
knownfaces = [list of student's names]
face_names = []
rgb_frame = frame[:, :, ::-1]
face_locations = face_recognition.face_locations(rgb_frame)
face_encodings = face_recognition.face_encodings(rgb_frame, face_locations)
for face_encoding in face_encodings:
    match=face_recognition.compare_faces(known_faces, face_encoding,
                                          tolerance=0.50)
    face_names.append(name)
for (top, right, bottom, left), name in zip(face_locations, face_names):
    cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)
```

3.5 Uploading Attendance Data to the Cloud

The recorded attendance is then stored as JSON data and is uploaded to firebase using the Python-Firebase library [9] which takes JSON data as well.

Function 3.5: *Program that uploads a single attendance to the database.*

```
import datetime
from firebase.firebase import FirebaseApplication, FirebaseAuthentication


if __name__ == '__main__':
        SECRET = '12345678901234567890'
        DSN = 'https://firebase.localhost'
        EMAIL = 'your@email.com'
        authentication = FirebaseAuthentication(SECRET,EMAIL, True, True)
        firebase = FirebaseApplication(DSN, authentication)
firebase.get('/users', None,
        params={'print': 'pretty'},
        headers={'X_FANCY_HEADER': 'very fancy'})
data = {'name': 'Ayush Singh', 'RegdNo': 1641012214,
                'recorded_at': datetime.datetime.now()}
                snapshot = firebase.post('/users', data)
                print(snapshot['name'])
def callback_get(response):
        with open('/dev/null', 'w') as f:
                f.write(response)
                firebase.get_async('/users', snapshot['name'],
                                        callback=callback_get)
```

4 Results and Analysis

After taking the snap of the classroom, we pass the image through the model once which gives us all the instances of the humans present in the room, and the masks which are the output of the model are used to separate the humans from the surroundings. Each of the separate instances of human is then fed to the facial recognition model which detects faces in the picture provided; if any faces are detected, we proceed to recognize the face and record the attendance.

This way, our aim of automating the attendance system in a class was achieved with an accuracy of almost 80% in real time. The accuracy is not promising, but we aim to improve it in the near future. The face_recognition model proves to be more robust when we use the model on a snap of a classroom as MASK-RCNN helps us to separate different entities in the image; however, it is not feasible enough for real-time facial recognition.

We can see that this is a better alternative to biometric attendance systems if done correctly and if the proper resources and computation power are provided.

5 Conclusion

To conclude, the primary aim of recording the attendance without any manual intervention is achieved on an almost satisfactory level; however, this application will also help a teacher to better understand the interests of his students and help them learn and focus better in a class.

We plan to implement the following in our application in the future:

- We plan to implement pos2seg [10] in our application to better predict the behavior of the students and achieve higher accuracy while creating a mask of the students
- We aim to make the program perform on real-time speeds so that it can receive more input data which will further improve the accuracy of the models.
- We plan to make the whole application feasible for low devices with low computing powers, hence making it less resource intensive.

References

1. Josphineleela, R., Ramakrishnan, M.: An efficient automatic attendance system using fingerprint reconstruction technique
2. Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from RGBD images. arXiv:1107.0169
3. Bolya, D., Zhou, C., Xiao, F., Lee, Y.J.: YOLACT: real-time instance segmentation. https://github.com/dbolya/yolact/
4. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN for object detection and segmentation, 24 Jan 2018. arXiv:1703.06870
5. Dlib: A toolkit for making real world machine learning and data analysis applications in C++. https://github.com/davisking/dlib
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv:1512.03385
7. Geitgey, A.: Facial recognition. https://github.com/ageitgey/face_recognition
8. Matterport, Inc. https://github.com/matterport
9. Python interface to the Firebase's REST API. http://ozgur.github.io/python-firebase/
10. Zhang, S., Li, R., Dong, X., Rosin, P., Cai, Z., Han, X., Yang, D., Huang, H., Hu, S.M.: Pose2Seg: detection free human instance segmentation 8 Apr 2019. arXiv:1803.10683

Website on Smart Project Management System by Application of Scrum Technology

Ankita Satapathy, Sweta Snigdha Sahoo, Harshita Raj, Yash Gupta, and Satyabrata Maity

Abstract Scrum is a project management system that prioritizes the work that matters the most and breaks it down into manageable chunks. Scrum itself is a simple framework for effective team collaboration and also for complex products. It is recognized as an effective and efficient device that is used for software development. It further includes definitions and specifications, especially regarding certain software development practices. It is a widely adopted system for achieving software agility. This is a project designed to guide a cluster of members in an iterative and incremental function of work. This also allows the user to trigger and make an update to the scheduled state for any kind of drag and drop action. It focuses on the use of an experimental process that allows a particular group to respond quickly in a systematic manner. Also, the decomposition of the system into certain structures can help to instruct some specific parts of the system.

Keywords Scrum methodology · Scrum software development · HCI · Web development

A. Satapathy · S. S. Sahoo (✉) · H. Raj · Y. Gupta · S. Maity
Computer Science and Engineering, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha 751030, India
e-mail: swetavsv29@gmail.com

A. Satapathy
e-mail: ankita123sat@gmail.com

H. Raj
e-mail: harshitaraj248@gmail.com

Y. Gupta
e-mail: yashgupta9658@gmail.com

S. Maity
e-mail: satyabratamaity@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_68

1 Introduction

Scrum involves constant evaluation and revaluation of the progression of a project, to ensure whether it is completed on time or not. It also provides an empirical chart to track a product's progress through all the stages of its development. So basically, here we are building a website which is all about collaborating and communicating both the people who are doing the work and the people who need the work to be done. If we apply scrum for any system, then it will help for the rapid growth of that system. It is a process for providing visibility to the teams who are working on some specific projects.

This piece of work helps in shifting data from one column to another column based on their functionality. This also helps us to recognize updates in the work process and work progress as well. It will help the members to organize people joining any group and will make the work easier for them. It will also let them know who is joining the group, who is leaving the group and what is their present status in the group. This website will show the next few tasks that need to be completed with its current state of progress providing an overview of the scrum and its applications.

Without the scrum management

1. There will be conflict during work functionality.
2. It will make the workflow complex and difficult for each one of the members to understand.
3. It will also lead to time complexity.
4. And will also be difficult to point out whether the person is active in the group or not.

Web development broadly refers to developing websites through the Internet or Intranet. It may further include the scripting of the client side and server side along with some features like designing of web and some other tasks. A complete layout of web development may include the full stack which can further be distributed into two streams that is the front end and the other is the back end. In our project, in the front end, we have used js and bootstrap, and in the back end, we have used MongoDB as the database, i.e. NoSQL database along with node.js framework.

2 Related Works

2.1 Note Log

This is a website that updates multiple entries in each note. Whenever a new entry is being selected for a particular note, then automatically the current date, time and location get tagged in. Best way is to resolve complex problems and to track the idea [1].

2.2 Tangled Web

It is a platform that creates a link between users' data. It matches the words and phrases with the data [1].

2.3 Monday.com

It is a platform that helps to manage teams and projects. It was a tough time tracking capabilities and notifications that can be customized and allows us to point out what is important. We can visualize and keep everyone in sync [2].

2.4 Yodiz

This is a versatile and robust enough to use in various cases. It is best for the very small team as it includes 100% functionality as well as features when there are three users. It is completely free for nonprofits, NGOs and universities. The makers of Yodiz provide community access to the strength of agile development [2].

2.5 Workflow Management System for MNC

This is an objective helpful of managing the workload of an employee which consists of employee's working details, leave details, designation details, experience details and software project details [3].

3 Proposed Work

In any website, there is a chance of disappearing of data if not organized in a well-planned manner. But with the application of scrum, we can ensure to secure the data. With the application of scrum's concept, we can reduce the clumsiness arising between a bunch of data. Without this scrum project, it will be difficult to handle the tasks in a well-organized manner which will further result in a clustering of people and project ideas in an unstructured methodology.

A product is always being prepared by the owners of the product. The steps taken are determined by the scrum master using its features. While we determine the backlog, the master of scrum is held by the manager who will handle the project.

There is a meeting of all members to make a plan of what is to be done. After we determine the product backlog, we go on discussing the purpose and goal of every feature as said by the owner. When we plan about the sprint, then the members of the team will detect the total duration each member will spend on each feature. The people should choose, understand and explain the tasks that need to be carried out.

There should be a daily report by a meeting about what each person has done. Here at this point, all the members of the team cannot be present but the members who are specifically working on the project must have their presence. At the end, when everything is completed, all of the members should know what is needed to be pursued in the rest of the task given.

3.1 Scrum Model

The steps for the study start with the preparation of a developer to broaden the area of study for a deeper knowledge of the methods involved in a project. The research on this project also gives information about the scrum framework and how to implement it on the development of the software and its project management and work on it using an application introduced as red mine.

3.2 Propounded Model

1. We can add security to this website to maintain the privacy of student data and not let it go to some other fake websites. Here, the algorithm Secure Hash Algorithm (SHA) secures the password.
2. We can further proceed with "Dynamic Work Distribution". It supports the division of workflow to perform any operations based on some specific tasks. It also provides projects to the people who are having no tasks at all by using scheduling sort algorithm.
3. It will be smartly checking the submissions done by the students and will also auto-update all the jobs done by some users at a particular period. They will also be utilizing their submitted projects and ideas for some future use.
4. To examine whether two students are simultaneously working on the same project at the same time or if a team is working on an assigned project.
5. Feature for the assignee to add comments on projects which he/she has assigned and keep track of the progress status and on the other side, a feature for students to add issues to their project to let know others if they could help them to modify.
6. Socket enables us to trade information between two machines for work distribution which allows access to consolidate data.
7. A function "divulge" would collect the history of projects that someone wants to contribute on and will help to validate confidentiality.

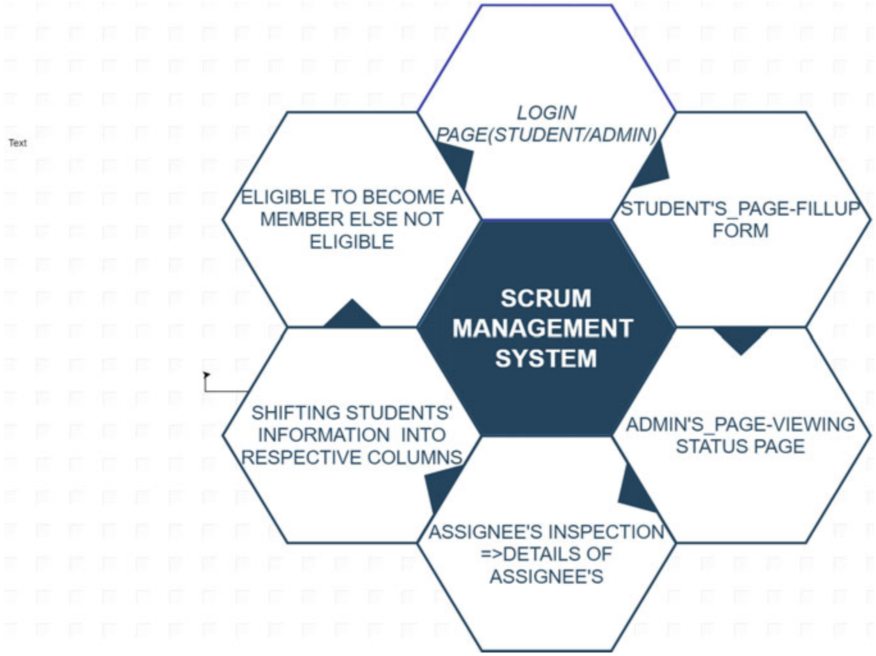


Fig. 1 Circular flow diagram of scrum model

The logic behind the system proposed is shown in Fig. 1. This tells the ones to need to prepare a front end of the program and also displays back-end needs. The specific task was representing the work that will be proposed.

3.3 Smart Scrum

Future work will be adding HCI and cloud computing for management of work. Human–computer interface (HCI) is a technique that studies the interaction of person and the computer. It deals with working, execution and evaluation of computer systems that have made up for human readability, and in concern of specific fundamentals of an interface, it enhances the workflow. Human–computer interface uses designer techniques for evaluation and software prototyping to review the scrum interfaces, etc.

The main perspective is the excellence in quality of products, reliability, improved efficiency in development and fewer errors result from a greater drive over review periods. The rapid increase in the acquaintance of the scrum method proves its effectiveness, efficiency, belief and usefulness. By using a network in the system, its needs are passing on directly. Scrum applications allow businesses to access data and share private data securely and assign the task given to the crew, which will be managed with the key advantages of cloud computing. Adding cloud to the built-in system is an idea for greater implementation of a small idea to create ease of access between the co-workers. The cloud results in greater performance and high scalability and accessibility to manage data. It provides permissions for the specific task to a specific group of teams.

3.4 Algorithm

1. Create your account by providing the requested information.
   (a) In the Created list, wait for a task to be assigned to you by Crew Head.
   (b) As the task has been assigned, complete within the deadline.
   (c) When you are done, let your assignee inform if there are any modifications.

      if (they accept it)

         You are a member of the team.

      else

         You are in the Default list and your id will be banned.

2. Admin work is to keep a record of all task that has been assigned to a new worker and under whom it has assigned and also to track on:
   (a) The work's (assignment provided) progress.
   (b) Database Management.
3. Crew Head will be assigned as the host of the website and will be able to track anywhere the progress of subgroups.

4 Results and Analysis

See Fig. 2.

4.1 Major Role

Here it goes into three perspectives: owner of the product, team and scrum master. The owner of the product is responsible for checking the specifications and the software applications which are to be built. They will list all the initials which are required to be done by the members of a team (known as the product backlog). It is to be done by a team where everyone will be responsible for completing their tasks of such

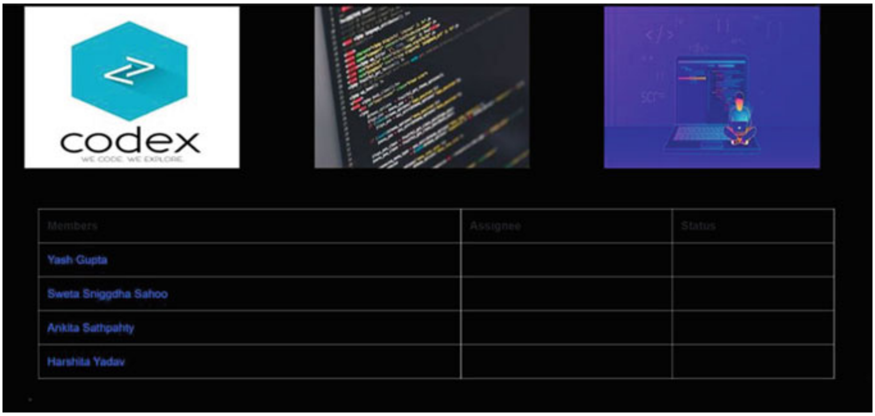


Fig. 2 Result representing the information of all the students who have added their data and are assigned with a project

backlogs as provided by the owner of the product, and the work has been divided on knowing what to do further. Scrum master always sets the process of a scrum during the project. Scrum master will also give an introduction and implementation on how the project of scrum works and guides all others for the assurance of understanding of the scrum method (Fig. 3).

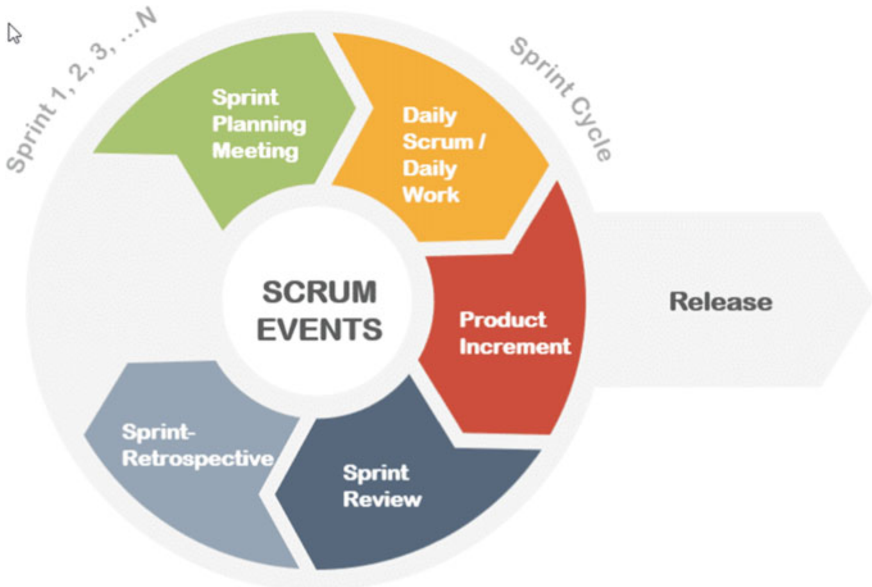


Fig. 3 Scrum cycle for describing the events included in a specific model

It is an idea that describes what to build and how to build systematically. This cyclic procedure helps us to manage work and to reach the desired prospect. These steps help in the creation and providing the right features and experience to the users.

4.2 Flow of Scrum

People can depict a project with the scrum of this system to be done at its best. After that, the owner of the project known as project owner plans an agenda for utilizing project backlog. We know that there is a list of plans to be done by a team which is known as a product backlog. There is a word known as sprint which signifies the plan that has to be executed in upcoming scrum sprint (29–30 days further). A sprint always begins with a sprint meeting planning which determines the work to be done in those next few days. Each team assembles and thinks upon their agenda and has a discussion as to "What has been done since the last daily scrum meeting?", "What problems are encountered during the work?", "What will be done for the next scrum?". Finally, there will be a meeting for 4 h duration which will be held by the scrum master to display a project or do a demo and finalize all the stuff that has been done till now.

5 Conclusion

Our project has gained rapid velocity through progressing and enhancing knowledge and minimizing the inspection and lengthy hard-working process. It has been benefited to provide a secure environment to work on with the easy flow of the task. As we were progressing towards the artefacts, it concluded with minimization of the huge problem into sub-scrum for us too. Giving exposure to a company's work in any industry would be harmful as it contains data that must be protected. Scrum has its own key information to maximize reliability and has transparency for ease.

Due to the creation of usable software, the time has shortened to create a product framework globally. The product working or system development or segmentation analysis is being done by the company through a concept or the need of a client and ends up with system deployment and operations. The continuous shifts and competition have changed in customer's needs and ensures the scrum policy, in which a system is required to change so as to maintain the need that is initiated by the customer for the future in the development stages of the working environment.

References

1. Grispos, G., Glisson, W., Storer, T.: Enhancing security incident response follow-up efforts with lightweight agile retrospectives. Digit. Invest. Int. J. Digit. Forensics Incident Response **22**:C, 62–73 (1-Sep-2017). Online publication date: https://dl.acm.org/doi/book/10.5555/1983547
2. Management and development of projects. https://thedigitalprojectmanager.com/managing-development-tools/
3. Visual overview of scrum. https://www.mountaingoatsoftware.com/agile/scrum/resources/overview

An Efficient and Secure Approach to the Traffic Fine System Using Hyperledger Fabric

Nehal Kumar Singh, Nishant Choudhary, and Rourab Paul

Abstract Our current, mostly analog system for taking traffic fine and validating documents or certificates (e.g., RC paper, driving license, etc.) is sluggish, complex, and unpredictable. Moreover, recently proposed traffic fine ordinances give rise to bribery. To ensure traffic fine and document verification as a transaction at real-time cost and in a hassle-free way, this paper proposes the use of a permissioned model of blockchain, i.e., hyperledger fabric technology for executing smart contracts. This paper also describes a prototype for carrying out the traditional fine in an efficient, secure, and transparent manner using hyperledger composer. Digitized strategy for validation of documents before each transaction and permissioned technology could supplant the traditional traffic fine system.

Keywords Blockchain · Hyperledger fabric · Hyperledger composer · Traffic fine system · Security · Privacy · Scalability · Efficiency

1 Introduction

Most of us have been there, you have just caught by the traffic police for not wearing a helmet and not having all the documents and certificates required such as driving license, RC paper, etc., and now you are dreading the time and effort it is going to take to deal with the police and other proceedings. Even if things go relatively smoothly, it is still a major disruption for the traffic. Blockchain specifically hyperledger fabric

---

N. K. Singh (✉) · N. Choudhary · R. Paul
Computer Science and Engineering, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha 751030, India
e-mail: nehalksingh@outlook.com

N. Choudhary
e-mail: nishantmailbox2011@gmail.com

R. Paul
e-mail: rourabpaul@soa.ac.in

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0_69

presents a tremendous opportunity for the traffic fine system. It provides the possibility to innovate around the way data is exchanged, requests are processed, and fraud is prevented. Blockchain technology provides an opportunity for the Government to improve operating efficiency, lower the costs of transaction processing, enhance the customer experience, improve the data quality, and increases trust among parties.

Today, the world is heading toward digitization in all aspects. Blockchain technology is forthcoming technology and is said to be one of the most desirable technologies which would transform the current circumstances of the world. In today's generation, use cases of blockchain are observed universally in every field such as financial market, supply chain, identity management, cross-border payment, cryptocurrencies, and many more such implementations of blockchain are being used [1]. The reason why blockchain is used in such varieties of field is due to its characteristics that include its both centralized and decentralized nature (centralized in hyperledger fabric and decentralized in Ethereum), transparency, consistency and security of data, immutability, non-corruptibility, low cost, and speed. The utilization of blockchain in Government sector such as traffic fine management can cause extensive changes in the administration and management. It can diminish corruption and misinterpretation by providing transparency in every fine collection transactions.

In this paper, we will be explaining the basics and applicability of blockchain and the effectiveness of blockchain technology for taking traffic fine, sharing of data and certificates required for verification, with the police using blockchain specifically hyperledger fabric. If a driver applies for a driving license, several procedures go on and on and result in corruption, and it is time-consuming as well and plenty of factors come into the picture. Moreover, if we do not take these situations into account and simply consider the way the traffic fine is being taken and the fine transaction is being recorded by the transport department in different parts of the country, we would find an unstable approach and inconsistent practice in the method and the blockchain technology can solve this problem. It can ensure quick fine collection and verification of data and certificates.

In the next segment of this paper, we have discussed the basics of blockchain technology after which in the third segment we have described our proposed solution to the traffic fine collection, data sharing and verification, and its working. After that in the fourth segment, we have provided a concise description of our prototype for the proposed solution that we have developed using the hyperledger composer.

2 Blockchain

Blockchain is a subset of distributed ledger technologies (DLTs), which builds a chronological chain of blocks, hence the name blockchain. According to Wikipedia definition, a blockchain is a growing list of records, called blocks, which are linked using cryptography [2]. Every block includes a cryptographic hash of the previous block, a time stamp, and transaction data. Time-stamping is a key characteristic of blockchain technology. Every block is time-stamped, with every new block referring to the previous block, and joined with cryptographic hashes, this time-stamped series of blocks stores an immutable record of all the transactions in the network, from the initial block [3]. One more important key feature of blockchain is the immutability of data, i.e., once a transaction is recorded on the blockchain, no one can modify/replace it, or at least, it would be extremely complex for anyone to modify it [4]. Smart contract is the program which is responsible for the execution of the predefined actions when certain condition within the system is met.

3 Proposed Solution

The fine transaction process can be simplified by the use of hyperledger fabric [5] as shown in Fig. 1. All the fine transactions submitted through a network are stored on the ledger. In the network, as shown in the figure, drivers can be associated with a unique identity, i.e., driving license and police can be associated with police id. The driver will propose a message to the police, and they will verify his/her signature and data and simulate the transaction by invoking chaincode to which the transactions refer. The figure shows how our model can be implemented in different states (like Odisha, Jharkhand, etc.). Every state will have a network, and every participant needs to be in the desired network to make the transaction. In every network, there will be the head of the department for keeping a view on all the transactions of the traffic, and above all, there will be a central authority like central point to all the states. The current set of assets (like driving license, insurance paper, RC, pollution certificate, etc.) are stored in the blockchain private state database called Side DB. The actual private data sent peer-to-peer via gossip protocol to only the organizations (in our case police headquarter) authorized to see it.

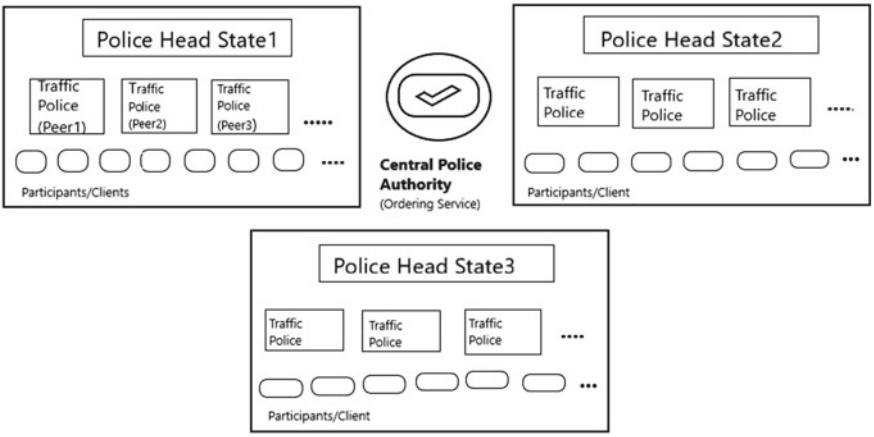


Fig. 1 Structure of the network

4 Prototype

Each network in our proposed solution defines

Participant: Driver, Police.
Transaction: Fine Given, Fine Received, Authorize Access, Revoke Access.
Event: Fine Transaction event, Access Transaction event.

Participants

- Driver: Drivers are the participants who are responsible for making the fine transaction if found guilty or for not following traffic regulations such as not wearing a helmet. Each driver participant is associated with a unique ID, i.e., driving license number. The driver requires to share all the documents or the certificates to the police headquarters before entering into the network of any state. If asked, the drivers need to make authorized access transactions as well to share their data with the police. After verification, they can make revoke access transactions for revoking access from the police.
- Police: Traffic police are the participants who are responsible for directing traffic and checking whether drivers are following the traffic rules. If a police participant catches a driver for not following the traffic regulations, he will ask that driver participant to make a fine given transaction and will allow the driver participant to leave. Now, for every fine given transaction that takes place, police in the headquarters will check all the data of that driver and will ask the driver to pay for all the violations of the traffic regulations. This will be online. Every police participant will have a unique ID, i.e., police identification number with which they will perform the fine received transaction if received fine from the driver. The transaction will be added to the public ledger.

Transactions

- Fine Given Transaction: The fine given transaction will be performed by driver participants using their unique ID when they will be asked from the police for not following or for violating the traffic regulation, e.g., not wearing a helmet. For executing this transaction, driver participants are required to fill three text fields: their ID, amount to be given, and the police ID of the police officer who caught them for infringing the traffic regulations. The successful transactions are added to the public ledger where everyone in the network can see the transaction occurred but the identity of the driver will be anonymous for all the other driver participants in the network.

Algorithm 1: Fine Given Transaction

```
Function pay(fineGiven)
    me = getCurrentParticipant()
    if me then
        event = FineTransactionEvent
        event.fineTransaction = fineGiven
        emit event
        update registry
        return me
    else
        return "Participant does not exist"
```

- Fine Received Transaction: The fine received transaction will be performed by the police participant if they receive a comprehensive amount from the driver participant who was asked to pay for violating the traffic regulations. For performing the transaction, the police participants need to fill three text fields: their ID, the amount that they have received from the respective driver participant, and the driver's license number. Once the transaction is successful, it is then added to the public ledger, and all the participants (police and the driver) can view the transaction but the identity of the police participant remain anonymous as the driver participants can only see the ID number and not the data of the police participants or the driver participants.

Algorithm 2: Fine Receive Transaction

```
Function Received(fineReceived):
    me = getCurrentParticipant()
    if me then
        event = FineTransactionEvent
        event.fineTransaction = fineReceived
        emit event
        update registry
        return me
    else
        return "Participant does not exist"
```

- Authorize Access Transaction: Authorized access transactions will be performed by driver participants for sharing their data with a particular police participant if asked by the police participant. Sharing of data can only be executed with the police participant having a valid police ID, and those police participants are required to be in the same network as that of the driver participant else the transaction will be rejected. Once the transaction is completed, it will be added to the public ledger, and the police participant can see the data of that driver from their wallet. Driver participants can revoke access to their data from the police participant by the revoke access transactions if their data gets verified.

Algorithm 3: Authorize Access Transaction

```
Function AuthorizeAccess(authorize)
    me = getCurrentParticipant()
    if me then
        if me.authorize then
            index = policeID
            me.authorized.push(authorize.policeID)
            event = AccessTransactionEvent
            event.accessTransaction = authorize
            emit event
            update registry
            return me
    else
        return "Participant does not exist"
```

- Revoke Access Transaction: Revoke access transactions will be performed by the driver participant for revoking access to their data from any police participant authorized previously. Revoking access transactions can only be accomplished if already authorized to the same police participant that means if the police ID matches the already authorized IDs else the transaction will be rejected. For revoke access, transaction drivers are required to give a valid police ID, and once the transaction is successful, the police participant who was previously able to view the data of that driver participant will no longer be able to view those data until that driver again authorizes that police participant to view his data by executing authorized access transaction.

Algorithm 3: Authorize Access Transaction

```
Function RevokeAccess(revoke)
    me = getCurrentParticipant()
    if me then
        if me.authorize == revoke.policeID then
            event = AccessTransactionEvent
            event.accessTransaction = revoke
            emit event
            update registry
            return me
    else
        return "Participant does not exist"
```

Event: The fine transaction event and the access transaction event are performed to emit fine transactions and access transactions, respectively. It is emitted in the transaction processor functions file. For the event to be published, the transaction which creates the event must call three methods, the first is the getFactory(), the second is factory.nextEvent(), and at last, it should be emitted by calling emit() function [6].

We examined our prototype in the hyperledger composer [7] playground by submitting a few transactions [8]. We created two driver participants and one police participant. We have created a fine given transaction from one of the driver participants and a fine received transaction from the police participant as shown in Fig. 2. We can view the complete details of the record like time stamp of the transaction, ID of the driver participant, and what amount driver participant had paid and who was the police in charge of that transaction through view record for every transaction but details are restricted to police, driver participant who did that transaction, and the network admin only. We have also performed the authorized access transaction (refer to authorize access transaction paragraph for more details) for evaluating whether the sharing of data is working perfectly or not as shown in Figs. 2, 3, and 4.

Date, Time	Entry Type	Participant	
2019-12-02, 14:35:35	AuthorizeAccess	9541 (Driver)	view record
2019-11-29, 22:43:46	AuthorizeAccess	9541 (Driver)	view record
2019-11-29, 22:38:33	AuthorizeAccess	9541 (Driver)	view record
2019-11-29, 22:34:36	FineReceived	6204 (Police)	view record
2019-11-29, 22:32:00	FineGiven	9541 (Driver)	view record

Fig. 2 Transaction registry after submission of transaction

Fig. 3 Data registry of participant

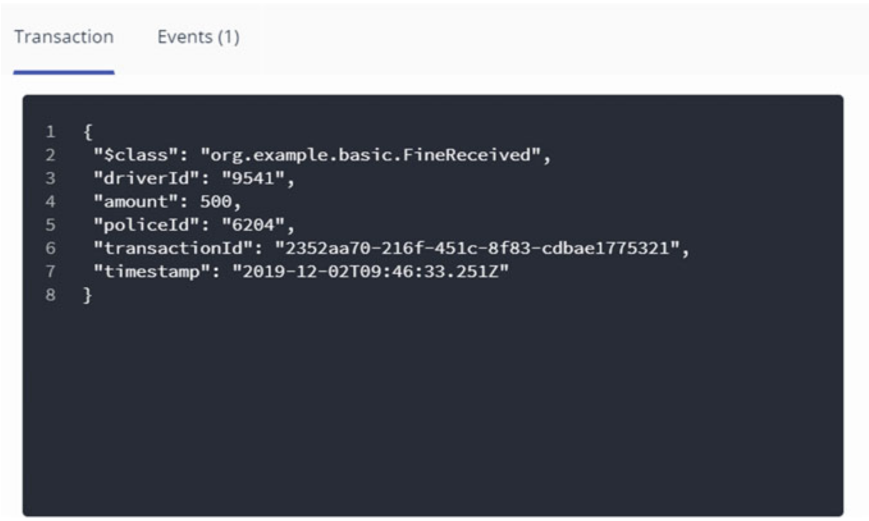


Fig. 4 Record with transaction ID and time stamp

5 Conclusion and Future Scope

The proposed solution can be used for an efficient fine transaction by fastening the process and also diminishes the time consumed in the verification of the documents or the certificates. It also makes the process transparent to avoid spurious allegations. The major benefit of this model is digitally encrypted credentials, granting permission and revoking permission from the police, easy transactions, and no dependability on the single participant [9]. This solution overcomes all the shortcomings of the current methodology by using the blockchain and makes the process more reliable. The hyperledger fabric is currently in an evolving stage, as far as the use cases, like traffic fine system, are concerned since the deployment levels have been progressing continuously. It is going to transform the current system by eradicating various defects at the same time. In this paper, we have only considered data and not documents or certificates but we hope that the documents or the certificates could be associated similarly thus creating a mesh, which would be better in terms of security and transparency for the system.

References

1. Mohite, A., Acharya, A.: Blockchain for government fund tracking using Hyperledger. In: International Conference on Computational Techniques, Electronics and Mechanical Systems, CTEMS (2018)
2. Wikipedia, Hyperledger Fabric. https://en.wikipedia.org/wiki/Blockchain. Accessed 31 Dec 2019
3. Kalra, S., Goel, S., Dhawan, M., Sharma, S.: ZEUS: analyzing safety of smart contracts. In: 25th Annual Network and Distributed System Security Symposium, NSDD (2018)
4. Sekhari, A., Chatterjee, R., Dwivedi, R., Negi, R., Shukla, S.: Entangled blockchain in land registry management. BW3, India (2019)
5. Hyperledger Fabric. https://hyperledger-fabric.readthedocs.io/en/release-1.4/. Accessed 31 Dec 2019
6. Cachin, C., Guerraoui, R.: Introduction to reliable and secure distributed programming. Springer (2011)
7. Hyperledger Composer. https://www.hyperledger.org/projects/composer. Accessed 31 Dec 2019
8. Composer Playground. https://developer.ibm.com/tutorials/cl-blockchain-basics-intro-bluemix-trs/. Accessed 31 Dec 2019
9. Varghese, J.: Building a secure platform to issue and verify digital documents. Print2Block, India (2018)

Author Index

D. Mishra et al. (eds.), *Intelligent and Cloud Computing*,
Smart Innovation, Systems and Technologies 153,
https://doi.org/10.1007/978-981-15-6202-0