

Mohit Kumar
Sukhpal Singh Gill
Jitendra Kumar Samriya
Steve Uhlig *Editors*

6G Enabled Fog Computing in IoT

Applications and Opportunities



Springer

6G Enabled Fog Computing in IoT

Mohit Kumar • Sukhpal Singh Gill •
Jitendra Kumar Samriya • Steve Uhlig
Editors

6G Enabled Fog Computing in IoT

Applications and Opportunities

 Springer

Editors

Mohit Kumar
Department of Information Technology
Dr. B.R. Ambedkar National Institute
of Technology
Jalandhar, Punjab, India

Sukhpal Singh Gill 
School of Electronic Engineering
and Computer Science
Queen Mary University of London
London, UK

Jitendra Kumar Samriya
Department of Information Technology
Dr. B.R. Ambedkar National Institute
of Technology
Jalandhar, Punjab, India

Steve Uhlig
School of Electronic Engineering
and Computer Science
Queen Mary University of London
London, UK

ISBN 978-3-031-30100-1

ISBN 978-3-031-30101-8 (eBook)

<https://doi.org/10.1007/978-3-031-30101-8>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

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

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Abstract

With the deployment of the 5G network, the role and applications of the Internet of Things (IoT) have increased tremendously in various domains, leading to the requirement as well as improvement in wireless communication systems. The superset of IoT is the Internet of Everything (IoE) which creates huge amounts of data by heterogeneous devices and needs the service of cloud and artificial intelligence (AI) for storage and intelligent processing. Nonetheless, the foundational and crucial elements of an IoE depend heavily upon the computing intelligence that could be implemented in the 6G wireless communication system. The IoT and 6G wireless communication networks are expected to transform customer services and applications and pave the way for completely intelligent and autonomous systems in the future.

Future 6G networks could benefit from the storage and computational services that fog computing offers. Fog computing, first used or coined by Cisco, is a decentralized infrastructure that locates processing components and storage at the cloud's edge, close to data sources like application users and sensors. Fog computing plays an important role in supporting the Internet-of-Things applications in the 6G network. It offers better security, saves network bandwidth, reduces latency, better privacy, and many more. The other popular technology paradigms covered in the book are cloud computing and artificial intelligence. Cloud computing offering the services to end users in the form of resources such as hardware, and software as per the user requirements pay per use basis. Artificial intelligence is the branch of engineering that makes machines or computer programs intelligent, in simple terms; AI enables machines to perform a task like a human. AI makes the cloud more secure, cost-effective, enables intelligent automation, increases reliability, and enhances data management.

This book covers the applications of fog enabled IoT networks with 6G endorsement in various domains such as healthcare, smart home, vehicular network, smart transportation, networking, and real-time businesses. At the end of the book, the reader will have knowledge of the concepts related to IoT applications, security, and

privacy issues in networking, and AI approaches to deal with them. The book also covers the role of machine learning in the advancement of 6G technology.

Keywords Sixth generation (6G) network; Fog computing; Internet of Things (IoT); Artificial intelligence; Blockchain; Intelligent environment; Mobile edge computing

Preface

The way machines and humans communicate has changed remarkably, which is a result of the rapid development of wireless communication networks. The increasing demand for a high data rate and the number of connected devices are reasons for such evolutionary developments. Fifth-generation (5G) network deployment is currently underway and giving end devices Gigahertz (GHz) connection. This will make life easier for all and have a big impact on how efficiently businesses operate. A new generation of cellular devices typically replaces an older model after about 10 years. By 2030, 6G is anticipated to be standardized and ready for deployment; therefore, research attention is beginning to shift to 6G communication systems. 6G networks are expected to support applications beyond traditional or current mobile use case scenarios, such as pervasive intelligence, virtual and augmented reality (VR/AR), Internet of Things (IoT), and ubiquitous instant communication. Novel wireless technologies and architectures will be explored in next-generation connectivity, having the benefits such as high data rates, new services, and ultra-low latency.

The advances in IoT resulted in the emergence and development of 5G and 6G communications respectively. Intelligent learning techniques used in IoT networks with 6G connectivity will enable the speedy completion of complex computations, revolutionizing user experience with nearly real-time responses. The increase in network capacity proportionately increases the network complexity. Numerous challenges that will be faced by 6G IoT include interoperability, scalability, quality-of-service (QoS) provisioning, heterogeneity, network congestion, integration, battery lifetime, and network capacity. To handle these issues, IoT will depend on rigorous deployment and intelligent learning techniques of fog computing devices that are placed closer to the end devices. By bringing computations to the end devices, fog computing devices will relieve the cloud server's workload and reduce computation latency. To significantly increase network efficiency, fog devices will intelligently incorporate idle resources from all available devices. Computation resources of fog devices will be the solution to address the demands of future applications.

The motive of this book is to capture the opportunities and applications of 6G communications to enable fog computing in IoT domains such as smart cities, industry, consumer applications, smart homes, and many more to explore.

Jalandhar, Punjab, India
London, UK
Jalandhar, Punjab, India
London, UK

Mohit Kumar
Sukhpal Singh Gill
Jitendra Kumar Samriya
Steve Uhlig

About the Book

The latest method for edge devices or mobile systems in terms of reducing energy usage and traffic congestion integrating with IoT device applications is sixth generation (6G) technology. The 6G network can be used in various platforms as an application, having a scope beyond 5G (B5G), such as intelligent techniques for software-defined networking (SDN), fog computing-enabled IoT networks, energy-aware location management, and 6G-enabled healthcare industry. This technology still has to face some issues with security and IoT-enabled trust networks. The practical and theoretical aspects of successfully implementing innovative intelligent techniques in a number of fields, such as fog computing, 6G, artificial intelligence, Internet of Things, and cloud computing, are explored in this book's emerging research. This book will serve as a source of inspiration for IT experts, academicians, researchers, industry professionals, authors, and engineers who are looking for current research on emerging perspectives in the area of 6G-enabled fog computing for IoT applications.

Contents

Part I Applications

AI Enabled Resources Scheduling in Cloud Paradigm	3
Sudheer Mangalampalli, Ganesh Reddy Karri, and Prabha Selvaraj	
Role of AI for Data Security and Privacy in 5G Healthcare Informatics ..	29
Ananya Sheth, Jitendra Bhatia, Harshal Trivedi, and Rutvij Jhaveri	
GPU Based AI for Modern E-Commerce Applications: Performance Evaluation, Analysis and Future Directions	63
Sanskar Tewatia, Ankit Anil Patel, Ahmed M. Abdelmoniem, Minxian Xu, Kamalpreet Kaur, Mohit Kumar, Deepraj Chowdhury, Adarsh Kumar, Manmeet Singh, and Sukhpal Singh Gill	
Air Quality Index Prediction Using Various Machine Learning Algorithms	91
Mann Bajpai, Tarun Jain, Aditya Bhardwaj, Horesh Kumar, and Rakesh Sharma	
Leveraging Cloud-Native Microservices Architecture for High Performance Real-Time Intra-Day Trading: A Tutorial	111
Mousumi Hota, Ahmed M. Abdelmoniem, Minxian Xu, and Sukhpal Singh Gill	

Part II Architecture, Systems and Services

Efficient Resource Allocation in Virtualized Cloud Platforms Using Encapsulated Virtualization Based Ant Colony Optimization (EVACO)	133
Nirmalya Mukhopadhyay, Babul P. Tewari, Dilip Kumar Choubey, and Avijit Bhowmick	

Authenticated, Secured, Intelligent and Assisted Medicine Dispensing Machine for Elderly Visual Impaired People..... 153
 Soubraylu Sivakumar, D. Haritha, S. Shanmugan, Talasila Vamsidhar, and Nidumolu Venkatram

Prediction of Liver Disease Using Soft Computing and Data Science Approaches 183
 Dilip Kumar Choubey, Pragati Dubey, Babul P. Tewari, Mukesh Ojha, and Jitendra Kumar

Artificial Intelligence Based Transfer Learning Approach in Identifying and Detecting Covid-19 Virus from CT-Scan Images 215
 Soubraylu Sivakumar, D. Haritha, Ratnavel Rajalakshmi, S. Shanmugan, and J. Nagaraj

Blockchain-Based Medical Report Management and Distribution System 239
 Subham Kumar Sahoo, Sambit Kumar Mishra, and Abhishek Guru

Design of 3-D Pipe Routing for Internet of Things Networks Using Genetic Algorithm 261
 Vivechana Maan, Aruna Malik, Samayveer Singh, and Deepak Sharma

Part III Further Reading

Intelligent Fog-IoT Networks with 6G Endorsement: Foundations, Applications, Trends and Challenges 287
 Syed Anas Ansar, Jitendra Kumar Samriya, Mohit Kumar, Sukhpal Singh Gill, and Raees Ahmad Khan

The Role of Machine Learning in the Advancement of 6G Technology: Opportunities and Challenges 309
 Krishna Kumar Mohbey and Malika Acharya

A Comprehensive Survey on Network Resource Management in SDN Enabled Data Centre Network 333
 Ashish Sharma, Sanjiv Tokekar, and Sunita Varma

Artificial Intelligence Advancement for 6G Communication: A Visionary Approach..... 355
 Javed Miya, Sandeep Raj, M. A. Ansari, Suresh Kumar, and Ranjit Kumar

AI Meets SDN: A Survey of Artificial Intelligent Techniques Applied to Software-Defined Networks 395
 Yadunath Pathak, P. V. N. Prashanth, and Ashish Tiwari

Editors and Contributors

About the Editors

Mohit Kumar is Assistant Professor in the Department of Information Technology at Dr. B.R. Ambedkar National Institute of Technology, Jalandhar, India. He received his Ph.D. degree from the Indian Institute of Technology Roorkee in the field of cloud computing, 2018, and M.Tech degree in Computer Science and Engineering from ABV-Indian Institute of Information Technology Gwalior, India, in 2013. He has received his B.Tech degree in Computer Science and Engineering from MJP Rohilkhand University Bareilly, 2009. His research topics cover the areas of cloud computing, fog computing, edge computing, Internet of Things, and soft computing. He has published more than 20 research articles in reputed journals and international conferences. He has been Session chair and keynote speaker of many International conferences, webinars, FDP, STC in India. He has guided three M. Tech students and guiding four Ph.D. scholar. He is an active reviewer of several reputed journals and international conferences. He is a member of IEEE.

Sukhpal Singh Gill is a Lecturer (Assistant Professor) in Cloud Computing at the School of Electronic Engineering and Computer Science, Queen Mary University of London, UK. Prior to his present stint, Dr. Gill has held positions as a Research Associate at the School of Computing and Communications, Lancaster University, UK, and also as a Postdoctoral Research Fellow at CLOUDS Laboratory, The University of Melbourne, Australia. Dr. Gill is serving as an Associate Editor in Wiley ETT and *IET Networks* journal. He has co-authored 70+ peer-reviewed papers (with H-index 30+) and has published in prominent international journals and conferences such as IEEE TCC, IEEE TSC, IEEE TII, IEEE TNSM, IEEE IoT Journal, Elsevier JSS/FGCS, IEEE/ACM UCC and IEEE CCGRID. He has received several awards, including the Distinguished Reviewer Award from SPE (Wiley), 2018, Best Paper Award AusPDC at ACSW 2021, and has also served as the PC member for venues such as PerCom, UCC, CCGRID, CLOUDS, IC FEC, and AusPDC. His research interests include cloud computing, fog computing, software

engineering, Internet of Things, and energy efficiency. For further information, please visit <http://www.ssgill.me>.

Jitendra Kumar Samriya is working as a Faculty at the Department of Information Technology, Dr. B.R. Ambedkar National Institute of Technology, Jalandhar. He has completed his Master of Technology and Ph.D. from BBA University (a central university), Lucknow. His research interests are cloud computing, artificial intelligence, and multi-objective evolutionary optimization techniques. He has published 15 research articles in reputed journals and conferences and published five Indian and international patents. He is also a member of IEEE and SCRS.

Steve Uhlig obtained a Ph.D. degree in Applied Sciences from the University of Louvain, Belgium, in 2004. From 2004 to 2006, he was a Postdoctoral Fellow of the Belgian National Fund for Scientific Research (FNRS). His thesis won the annual IBM Belgium/FNRS Computer Science Prize 2005. Between 2004 and 2006, he was a Visiting Scientist at Intel Research Cambridge, UK, and at the Applied Mathematics Department of University of Adelaide, Australia. Between 2006 and 2008, he was with Delft University of Technology, the Netherlands. Prior to joining Queen Mary University of London, he was a Senior Research Scientist with Technische Universität Berlin/Deutsche Telekom Laboratories, Berlin, Germany. Since January 2012, he has been the Professor of Networks and Head of the Networks Research group at Queen Mary, University of London. Between 2012 and 2016, he was a Guest Professor at the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. With expertise in network monitoring, large-scale network measurements and analysis, and network engineering, during his career he has published in over 100 peer-reviewed journals, and awarded over £3 million in grant funding. He is currently the Editor-in-Chief of ACM SIGCOMM *Computer Communication Review*, the newsletter of the ACM SIGCOMM SIG on data communications. Since December 2020, Steve has also held the position of Head of School of Electronic Engineering and Computer Science. Current research interests: Internet measurements, software-defined networking, content delivery.

Contributors

Ahmed M. Abdelmoniem School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Malika Acharya Department of Computer Science, Central University of Rajasthan, Ajmer, India

Syed Anas Ansar Babu Banarasi Das University, Lucknow, India

M. A. Ansari GBU, Greater Noida, India

Mann Bajpai Manipal University Jaipur, Jaipur, Rajasthan, India

Aditya Bhardwaj Bennett University, Greater Noida, Uttar Pradesh, India

Jitendra Bhatia Department of Computer Science & Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India

Avijit Bhowmick Department of Computer Science & Engineering, Budge Budge Institute of Technology, Kolkata, India

Dilip Kumar Choubey Department of Computer Science & Engineering, Indian Institute of Information Technology Bhagalpur, Bhagalpur, Bihar, India

Deepraj Chowdhury Department of Electronics & Communication Engineering, International Institute of Information Technology (IIIT), Naya Raipur, India

Pragati Dubey Department of Bioinformatics, School of Earth, Biological and Environmental Sciences, Central University of South Bihar, Gaya, Bihar, India

Sukhpal Singh Gill School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Abhishek Guru Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India

D. Haritha Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

Mousumi Hota School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Tarun Jain Manipal University Jaipur, Jaipur, Rajasthan, India

Rutvij Jhaveri Department of Computer Science and Engineering, SoT, PDEU, Gandhinagar, Gujarat, India

Ganesh Reddy Karri School of Computer Science and Engineering, VIT-AP University, Amaravati, India

Kamalpreet Kaur Seneca International Academy, Toronto, ON, Canada

Raees Ahmad Khan Babasaheb Bhimrao Ambedkar University, Lucknow, India

Adarsh Kumar Department of Systemics, School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India

Horesh Kumar Greater Noida Institute of Technology, Greater Noida, Uttar Pradesh, India

Jitendra Kumar School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu, India

Mohit Kumar Department of Information Technology, National Institute of Technology Jalandhar, Jalandhar, Punjab, India

Ranjit Kumar GCET, Greater Noida, India

Suresh Kumar GCET, Greater Noida, India

Vivechana Maan Department of Computer Science & Engineering, Dr. B R Ambedkar National Institute of Technology Jalandhar, Jalandhar, Punjab, India

Aruna Malik Department of Computer Science & Engineering, Dr. B R Ambedkar National Institute of Technology Jalandhar, Jalandhar, Punjab, India

Sudheer Mangalampalli School of Computer Science and Engineering, VIT-AP University, Amaravati, India

Sambit Kumar Mishra SRM University-AP, Amaravati, Andhra Pradesh, India

Javed Miya GCET, Greater Noida, India

Krishna Kumar Mohbey Department of Computer Science, Central University of Rajasthan, Ajmer, India

Nirmalya Mukhopadhyay Department of Computer Science & Engineering, Indian Institute of Information Technology, Bhagalpur, India

J. Nagaraj Computer Science and Engineering, Madanapalle Institute of Technology and Science, Madanapalle, Andhra Pradesh, India

Mukesh Ojha Greater Noida Institute of Technology, Greater Noida, Uttar Pradesh, India

Ankit Anil Patel School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

Yadunath Pathak Department of Computer Science & Engineering, Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, India

P. V. N. Prashanth Department of Computer Science & Engineering, Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, India

Sandeep Raj Ajeenkya D Y Patil University, Pune, India

Ratnavel Rajalakshmi School of Computing, Vellore Institute of Technology, Chennai, India

Subham Kumar Sahoo SRM University-AP, Amaravati, Andhra Pradesh, India

Jitendra Kumar Samriya Graphic Era Deemed to be University, Dehradun, India

Prabha Selvaraj School of Computer Science and Engineering, VIT-AP University, Amaravati, India

S. Shanmugan Department of Physics, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

Research Centre for Solar Energy, Department of Engineering Physics, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

Ashish Sharma Government Women's Polytechnic College, Indore, India

Deepak Sharma Department of Computer Science, Kiel University (Christian-Albrechts Universität zu Kiel), Kiel, Germany

Rakesh Sharma Manipal University Jaipur, Jaipur, Rajasthan, India

Ananya Sheth Department of Mathematics, St. Xavier's College, Ahmedabad, Gujarat, India

Manmeet Singh Centre for Climate Change Research, Indian Institute of Tropical Meteorology (IITM), Pune, India

Jackson School of Geosciences, University of Texas at Austin, Austin, TX, USA

Samayveer Singh Department of Computer Science & Engineering, Dr. B R Ambedkar National Institute of Technology Jalandhar, Jalandhar, Punjab, India

Soubraylu Sivakumar Computing Technologies, SRM Institute of Science and Technology, Chennai, India

Babul P. Tewari Department of Computer Science & Engineering, Indian Institute of Information Technology Bhagalpur, Bhagalpur, Bihar, India

Sanskar Tewatia Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

Ashish Tiwari Department of Computer Science & Engineering, Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, India

Sanjiv Tokekar IET DAVV, Indore, India

Harshal Trivedi Softvan Pvt. Ltd., Ahmedabad, Gujarat, India

Talasila Vamsidhar Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

Sunita Varma S.G.S.I.T.S, Indore, India

Nidumolu Venkatram Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

Minxian Xu Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

Part I

Applications

AI Enabled Resources Scheduling in Cloud Paradigm



Sudheer Mangalampalli, Ganesh Reddy Karri, and Prabha Selvaraj

Abstract Cloud Computing was evolved as one of the paradigm, which gives services to users in a utility-based manner. Services of cloud computing were extended to various fields and applications. Due to the enormous number of users, flexibility and easy to use nature of cloud paradigm, many of companies are trying to migrate towards cloud paradigm but from a cloud provider perspective it is a difficult to job to handle or schedule these heterogeneous workloads, which are coming onto cloud console. Therefore, it is important for a cloud provider to employ a task scheduling mechanism, which should be more proactive based on the nature of workloads coming onto cloud interface and how effectively they are scheduled onto suitable virtual resources. Many of existing scheduling algorithms used nature or bio inspired techniques to model schedulers as scheduling problem in cloud paradigm is a classical NP-Hard problem but still to make a schedule for a task onto a suitable VM based on its processing capacity while minimizing its makespan, energy consumption and other operational costs is still a tedious job as incoming user requests are highly dynamic in nature. In this paper, we have used a deep reinforcement learning technique i.e. DDQN model to make decisions of scheduling in cloud paradigm while checking incoming requests and underlying resources for every task. Task priorities are evaluated for all incoming tasks and prioritized tasks are fed to our scheduler and based on imposed conditions our scheduler will make decisions effectively. This entire research implemented on cloudsim. Extensive simulations are conducted by generating workload randomly and from realtime workload traces. Finally, our proposed scheduler is evaluated against existing baseline approaches i.e. Round Robin, FCFS, and Earliest Deadline first. From Simulation results, our proposed approach shown a huge impact over existing baseline approaches in terms of makespan, Energy consumption.

S. Mangalampalli (✉) · G. R. Karri · P. Selvaraj
School of Computer Science and Engineering, VIT-AP University, Amaravati, India
e-mail: sudheer.mangalampalli@vitap.ac.in; ganesh.reddy@vitap.ac.in; prabha.s@vitap.ac.in

Keywords Task scheduling · Cloud computing · Artificial intelligence · Deep reinforcement learning · Double deep Q-network model · Round Robin · FCFS · Earliest dead line first · Makespan · Energy consumption

1 Introduction to Cloud Computing

Cloud Computing is a distributed model, which gives on demand ubiquitous services from virtual resources as a service needed for cloud consumers based upon service level agreements. With the advent of huge generation of data from various resources, there is a huge pressure on IT firms to maintain applications with commodity hardware and trying to adopt cloud environment [1]. In on premises environment, organizations follows cluster or grid computing approaches where cluster computing follows a centralized computing architecture [2] and Grid Computing follows either centralized or distributed computing approach [3] which renders their services to their corresponding customers. In Cluster and Grid Computing environments, computing resources are fixed and they cannot scale automatically as per the on demand requirements of users. Cloud Computing paradigm gives users scaling facility to increase or decrease virtual resources as per the requirements of users. This entire paradigm based on service oriented architecture [4] through which virtual resources will be given to all users as services as per SLA made between cloud user and provider. The main characteristics of cloud computing paradigm are mentioned below [5].

1.1 Characteristics of Cloud Computing

Resource Pooling It gives cloud users a virtual resource from pool of resources as per SLA of customer. This is an important characteristic in this paradigm as many of users will access virtual resources and these resources should be provisioned automatically from cloud provider from pool of resources running at cloud provider and allocation of resources to users should not affect other users while provisioning resources.

On Demand Service It provides self-control for cloud customer/user for the application running in cloud environment. He/She can provision or deprovision resources based on need of the application.

Ease of Maintenance This paradigm provides a huge flexibility to their users as they don't need to maintain their applications as they do in on premises environment. Therefore, users can focus on their development of application improvement and business objectives as cloud provider will take care of maintenance in terms of updates, patching etc.

Scalability This is a key characteristic in cloud paradigm as in now a days many of applications need to increase or decrease their resources instantly. To do so, we need a special paradigm, which adapts to environment and provision or deprovision resources according to situation. Coming to scalability it is of two types in cloud paradigm i.e. Horizontal and Vertical scaling. Horizontal scaling is to increase or decrease entire virtual machine if workload of the application cannot be handled by existing infrastructure. Vertical scaling is about to increase or decrease a specific resource.

No Upfront Investment It is economical as cloud user need not invest money on resources unlike in on premises environment. Zero upfront investment is needed in this paradigm, as users don't need to pay for their operational and administrative costs as in on premises environment.

Measured Service It is to be used for both cloud user and provider as for all services which were provisioned from cloud provider end need to compute pricing and from cloud user end it is to be used for them to know what are different services they have used and amount of consumption for corresponding services. This will be mainly useful for generating bill for cloud user automatically.

Security It is one of crucial characteristic of cloud paradigm as users are deploying their applications in third party environment not at their on premise environment. Therefore, cloud providers will follow high standards of security at their end and they will create a replica of each data point in the cloud environment as an end point through which data can be restored if any file gets corrupted or any crash happens in cloud environment. Cloud paradigm uses IAM i.e. Identity and access management service as their primary security service, which can give users a high standard of security at different levels, based on the need of the application.

Automation This is an essential characteristic for cloud paradigm as from around the globe many of users requests wide variety of resources according to their application. All these requests need to be fulfilled by the cloud provider as per the demand of cloud user according to SLA. Therefore, to handle these kind of heterogeneous and diversified requests from various users for different types of needs and provisioning those resources from cloud provider on demand needs automation. All provisioning and deprovisioning should be done automatically. If it should be done automatically, underlying policies need to be defined by cloud provider for different resources in the cloud environment.

Resiliency It is an important characteristic for cloud computing paradigm as if any data, file or computing server goes down or any crash happens it will be recovered quickly with little downtime unlike on premise environment.

Availability In this model, all virtual resources to cloud user will be highly available as many users accessing their resources on demand without any hesitancy. To accommodate resources to all users seamless access to be provided by cloud provider and infrastructure to be resilient enough to handle requests from cloud

users. Cloud paradigm is having enough strength of resilient network and highly available and scalable resources as they are using virtualization.

Remote Access In this environment all resources to cloud users will be given as services on demand and these resources can be accessible from anywhere around the world. Therefore, it uses Internet to access resources in cloud paradigm. It allows users to work from anywhere to access their applications in cloud environment, which gives high availability, resilient network and seamless access to users.

Multitenant Architecture It is a primary characteristic in cloud paradigm through which, a resource is shared among several users as per SLA. When a single resource is shared among several users, there will be no conflicts among provision of resources as it is having an underlying software program managed by cloud provider and it is automated.

The below Fig. 1 represents characteristics of Cloud Computing.

1.2 Deployment Models of Cloud Computing

To adopt to cloud environment into on premises environment either they need to migrate their existing environment onto cloud computing infrastructure or they need to build their new applications on cloud environment. IT firms need to use certain deployment models for their application deployment in cloud environment. There are certain deployment models [6] named as Private, Public, Hybrid and Community cloud models. The below are primary deployment models of cloud paradigm which are represented in Fig. 2.

Public Cloud renders services to all cloud users who subscribed to services of a corresponding cloud provider on a paid basis. This deployment model can render services to all of its users around the world and users can access cloud services at any time around the clock in a seamless manner.

Private Cloud renders services to cloud users to a specific organization based on the subscription of services made with cloud provider. This deployment model helps cloud users to access their application with a secured channel and no other users out of that organization can access the services.

Hybrid Cloud It is a combination of both public and private cloud services rendered by a cloud provider to an organization in which some of the resources can be accessed publicly and some other resources have to be restricted to part of users in organization. Therefore, to handle this situation, organizations will subscribe to a model named as Hybrid cloud model, which gives resource access to users based on the restrictions mentioned by their organization.

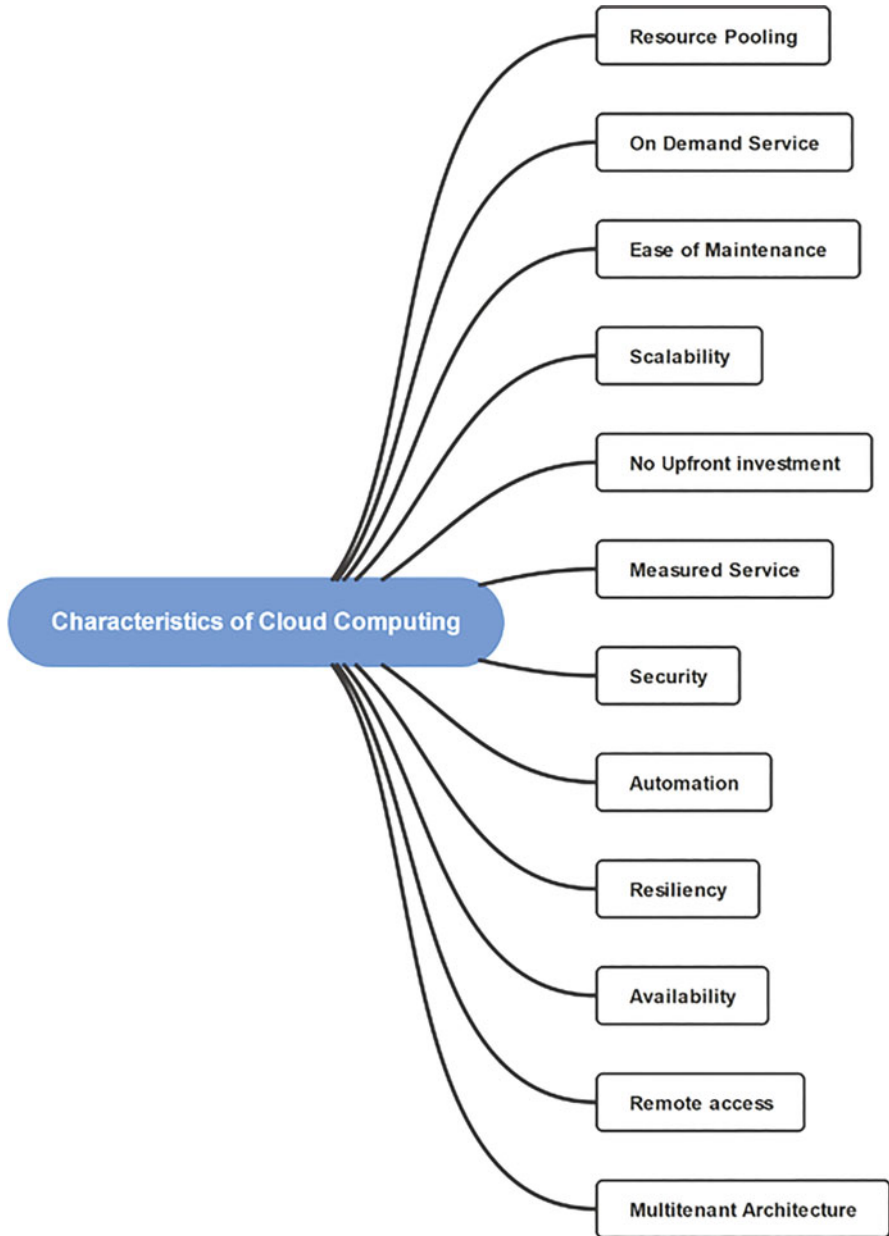
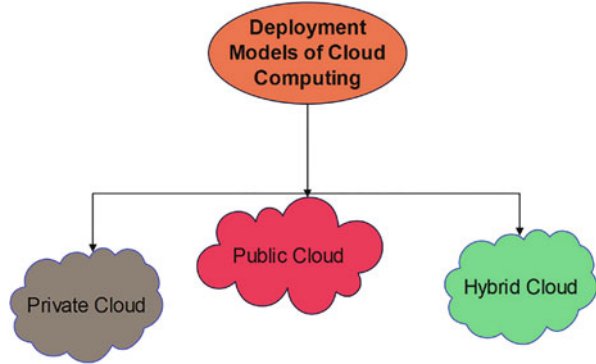


Fig. 1 Characteristics of Cloud Computing

Fig. 2 Deployment models of Cloud Computing



1.3 Service Models of Cloud Computing

Cloud Computing paradigm provides resources through service models as this paradigm renders resources to users based on SLA between cloud users and providers. There are several services available in these days with different cloud providers.

Infrastructure-as-a-Service This service model provides virtual infrastructure to cloud user, which can replace hardware infrastructure in on premise environment. The advantage of this service is to scale infrastructure to an extent based on the requirement of users. Entire infrastructure of user application can be accessed via user interface.

The below Fig. 3 represents basic service models [7] in cloud computing.

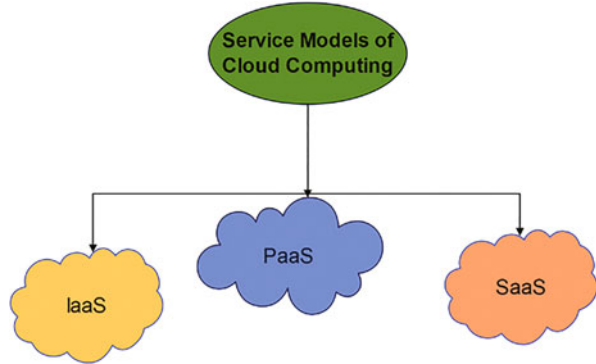
Platform-as-a-Service This model is used to provide virtual development opportunities for cloud user based on SLA between cloud user and provider. This model can be helpful to users to develop cloud naïve applications. It provides virtual development platform for users by integrating it with different environments with use of RESTFUL API service. It provides support of various languages and APIs to develop cloud naïve applications.

Software-as-a-Service This service model renders service to cloud users for applications hosted in cloud environment and developed by cloud provider or applications hosted in cloud environment and developed by other users. This model is not used to develop applications but to render services to users for applications already hosted in cloud environment.

Main highlights of this chapter are presented below.

- Main objective of this research is to design a task-scheduling algorithm using DDQN model i.e. reinforcement learning based approach.
- Energy consumption, makespan are considered as parameters in this approach.
- Workload considered from fabricated workloads and another real world dataset i.e. BigDataBench [36].

Fig. 3 Service models of Cloud Computing



2 Resource Scheduling in Cloud Computing

After discussion of basic service, deployment models we need to discuss about Provisioning of virtual resources to users by considering underlying resources in cloud infrastructure. This process is known as Resource scheduling [8]. In Cloud Computing, assignment of user requests to appropriate virtual resources is a highly dynamic scenario as many users are accessing cloud resources concurrently. It is a class of NP – hard problem and assignment of requests to virtual resources in cloud paradigm is a challenging issue as incoming requests coming onto cloud console varies with respect to time, size, processing capacity. Cloud paradigm needs an effective scheduler as provisioning and deprovisioning of resources to user requests is depends only on scheduler. Therefore, scheduler will impacts various parameters when assigning virtual resources to user requests. It will directly impacts the performance of cloud environment which impacts directly both cloud provider and users. Many of resource scheduling algorithms in cloud computing were modeled by nature inspired and metaheuristic algorithms. There are many existing algorithms i.e. PSO [9], GA [10], ACO [11], CSA [12], CSO [13] are used to model scheduling algorithms in cloud computing. There are many nature inspired and metaheuristic algorithms used to develop resource scheduling mechanisms in cloud paradigm but still it is a challenge in cloud paradigm to suitably map incoming tasks to appropriate virtual machines. Therefore, an artificial intelligence mechanism is needed to schedule incoming tasks to appropriate virtual resources. In this chapter, we used a DDQN model i.e. deep reinforcement learning technique to map incoming requests to VMs carefully based on incoming requests and checking underlying resources. Initially in this chapter, we carefully studied about various resource scheduling algorithms modeled by metaheuristic algorithms and their impact on cloud computing.

In the coming section, we mentioned various metaheuristic algorithms used to solve resource scheduling problems in cloud computing.

2.1 Resource Scheduling Algorithms Modeled by Metaheuristic Approaches in Cloud Computing

This section clearly discusses about various resource scheduling algorithms modeled by various metaheuristic approaches.

In [9], a scheduling framework designed by authors which focuses on minimization of task processing time. This algorithm modeled by using MPSO, which used over diversified population. It was simulated on Cloudsim. Main intention of this algorithm to minimize processing time of task with in deadline. It was evaluated against existing PSO, APSO, artificial bee colony, BAT, min-min algorithms. From simulation results it was proved that it was dominant over existing approaches for mentioned parameter. In [15], authors proposed a hybrid resource scheduling model was developed to address a single objective i.e. makespan. It was modeled by hybridization of whale algorithm by tuning parameters for both exploration, exploitation and to avoid premature convergence. It implemented on cloudsim. Workload taken from real world and synthetic datasets. It was compared over existing Whale optimization algorithm and finally evaluated makespan. From simulation results, it was proved that hybrid approach improves makespan over existing baseline algorithm for mentioned parameter. In [16], authors focused on development of a resource-scheduling model, which minimizes energy consumption, execution cost. Methodology used in this approach CSSA algorithm. Cloudsim was used as simulation tool. Real time and synthetic workloads given as input to algorithm. It compared over existing baseline models i.e. Hybrid GA-PSO, PSO-BAT, SSA algorithms. Results demonstrated that proposed model minimizes specified parameters. In [17], a resource scheduling model was developed to focus on parameters i.e. makespan, throughput, degree of imbalance. Hybrid Gradient was added to cuckoo search to solve resource-scheduling problem. Cloudsim was used as a simulation tool for experimentation. Real time work log traces from HPC2N [13], NASA [13] were used in simulation. It was compared over existing approaches i.e. ACO, CS. From results, it proved that HGDCS outperforms over existing algorithms for mentioned parameters. In [18], a resource-scheduling algorithm developed for vehicular cloud, which addressed makespan, energy consumption. It was mainly developed to schedule tasks properly onto vehicular clouds to avoid latency from centralized cloud architecture. It was modeled into three layers of scheduling and schedules on demand requests of road side users successfully based on usage of HAPSO by combining genetic, PSO algorithms. Sumo, NS2, MATLAB were used as simulation environments for resource scheduling. It was evaluated against PSO, GA algorithms. From simulation results it was greatly improved makespan, energy consumption by 34%, 32.5% respectively. In [19], task-scheduling mechanism was developed to focus on makespan. It works with crow search which is a nature inspired algorithm based on food habits of the crow. It was simulated on cloudsim. Heterogeneous random workloads were given as input to algorithm. It was evaluated over existing ACO, Min-Min algorithms. Simulation results revealed that existing works were outperformed by this approach

for mentioned parameter. In [20], authors formulated a task scheduling approach focused on parameters i.e. makespan, resource utilization, cost. LOA was used as methodology to solve scheduling problem. Cloudsim was used for entire simulation. Random generated workload used in simulation. It was evaluated over existing GA, PSO algorithms. Simulation results proved this approach was dominant over existing mechanisms for specified parameters. In [21], scheduling algorithm formulated to address computational cost, makespan, resource utilization, degree of imbalance. This mechanism modeled based on CSSA algorithm, which selects search space for optimization by using randomized inertia weights. It helps to converge swarm towards solution quickly. It implemented on cloudsim. It compared over GA, PSO, ABC approaches. It outperformed over existing algorithms for mentioned parameters. In [22], authors designed two scheduling algorithms i.e. LJFP, MCT based on existing PSO algorithm. It developed based on PSO in which modification was done at initialization of population done by these two-mentioned LJFP, MCT. It implemented on MATLAB. It was evaluated against baseline approach i.e. PSO. From results, it proved that these two approaches dominant over classical PSO in terms of execution time, energy consumption, degree of imbalance. In [23], a scheduling framework designed in two folds aimed at minimization of makespan, energy consumption. This approach based on scheduling tasks in compute clouds. Hybrid methodology used to solve task scheduling problem by combining GA, BFA. In first fold, this approach was addressed makespan. In second fold, it was addressed energy consumption. Entire simulations conducted on MATLAB. Metrics addressed in this approach evaluated with different workload heterogeneities. Initially simulation carried out with low heterogeneity workload and later it carried out with high heterogeneous and diversified workload. It was evaluated over existing GA, PSO, BFA algorithms. Simulation results revealed that it outperforms all existing approaches for specified parameters. In [24], SACO was developed to address how makespan, processing time of tasks effects scheduling in cloud computing. This algorithm uses diversification, reinforcement approach to avoid long path to capture their food shown by leader ants. Simulation carried out on Cloudsim. It was compared over variants of ACO. From results, it proved that SACO outperformed other variants for mentioned parameters. In [25], BMDA was proposed by authors to solve scheduling in cloud computing. Methodology used in this algorithm is a combination of BBO and dragon fly algorithms. BBO used as a technique to avoid premature convergence, which combined with dragon, fly algorithm to give optimal solutions. It implemented on Cloudsim. Workload given to this algorithm is from NASA [13] parallel work log archives and from CEC 2017 [26] benchmark functions. It compared over DA, PSO, BAT, GWO, RRO, Adaptive DA algorithms. From results, BMDA outperforms over existing algorithms for metrics i.e. response time, execution time, SLA violation. In [27], authors developed a task-scheduling algorithm focuses on minimization of makespan, maximization of resource utilization. EMVO developed by combining MVO, PSO algorithms that addresses local optimization problem in PSO. In EMVO, experiments conducted by using fixed and variable number of VMs. Entire experiments conducted on MATLAB. It evaluated over MVO, PSO algorithms and results revealed that it

shows huge impact over existing approaches for specified parameters. Authors in [28] proposed a task scheduling algorithm i.e. IE-ABC a hybrid metaheuristic approach addressed parameters i.e. Security, QoS. It was modeled by classical ABC approach which improved by adding a dedicated employee bee which keeps track of VM and datacenter status. Therefore, it is easy for a scheduler to look at VM and datacenter status to map its tasks easily and precisely. Simulations conducted on cloudsim. It was compared against classical ABC algorithm with respect to makespan, cost, Number of tasks migrated. Finally, from simulation results, there is a huge impact over existing ABC for specified parameters. In [29], scheduling algorithm formulated to schedule tasks onto virtual machines. CRO and ACO combined to solve scheduling problem. It implemented on Cloudsim. Random workload and Amazon EC2 instances workload given input to algorithm. It evaluated against CRO, ACO, PSO, CEGA algorithms and results revealed that it shows improvement in makespan, cost. In [30], FA-SA algorithm proposed by authors to introduce a new local search to optimize solution. This algorithm initializes a new population strategy to converge towards near optimal solutions. Cloudsim. used for simulation. Workload given to algorithm from real time datasets and synthetic workloads. It compared against existing firefly, SA, min-min, max-min algorithms. Results revealed that it shown huge impact over existing approaches for specified parameters makespan for different workloads with different datasets. In [31], a hybrid algorithm proposed by authors to schedule tasks effectively by addressing energy consumption, SLA violation. Methodology used in this algorithm is BMW-TOPSIS to map tasks to VMs. Entire simulations conducted on Cloudsim. It compared over existing BMW, TOPSIS algorithms and performed ANOVA test to evaluate statistics from results. From simulation results, it outperforms existing approaches for energy consumption, makespan, resource utilization.

Table 1, it clearly shown that many of metaheuristic algorithms addressed baseline parameters but scheduling in cloud computing environment is highly dynamic and to map tasks effectively and these metaheuristic approaches still facing challenges to get optimal solutions in terms of metrics addressed in cloud environment. Therefore, it is necessary to employ a machine-learning model in scheduling architecture through which decision need to be taken for mapping of requests with resources.

In the next section, we mentioned various ML based scheduling algorithms to solve resource scheduling problems in cloud computing.

2.2 Resource Scheduling Algorithms Modeled by Metaheuristic Approaches in Cloud Computing

This section clearly discusses about various resource scheduling algorithms in cloud computing modeled with various machine-learning techniques.

Table 1 Summary of metaheuristic resource scheduling algorithms in Cloud Computing

References	Methodology	Objectives of resource scheduling algorithms modeled by metaheuristic approaches
[9]	MPSO	Task processing time
[10]	Improved GA	Execution time
[11]	MORA-ACS	Energy consumption, load balancing
[12]	MOCSSO	Completion time, execution cost
[13]	CSO	Energy consumption, makespan, total power cost, migration time
[15]	Hybrid Whale	Makespan
[16]	CSSA	Energy consumption, execution cost
[17]	HGDCS	Makespan, throughput, degree of imbalance
[18]	HAPSO	Makespan, energy consumption
[19]	Crow Search	Makespan
[20]	LOA	Makespan, resource utilization, cost
[21]	CSSA	Computational cost, makespan, resource utilization, degree of imbalance
[22]	LJFP, MCT	Execution time, energy consumption, degree of imbalance
[23]	GA-BFA	Makespan, energy consumption
[24]	SACO	Makespan, processing time of tasks
[25]	BMDA	Response time, execution time, SLA violation
[27]	EMVO	Makespan, resource utilization
[28]	IE-ABC	Security, QoS
[29]	CR-ACO	Makespan, cost
[30]	FA-SA	Makespan
[31]	BMW-TOPSIS	Energy consumption, makespan, resource utilization

In [32], authors proposed an automation approach for scheduling workloads in cloud paradigm. Initially authors used three ML models to develop this scheduling algorithm i.e. RL, DQN, RNN-LSTM, DRL-LSTM. From all these approaches, DRL-LSTM works well in minimization of CPU usage cost, Memory usage cost. It was implemented using Pytorch framework. It evaluated against existing RR, SJF, IPSO algorithms. From results, DRL-LSTM shows a huge improvement in minimization of CPU usage cost 67% to SJF, 35% to RR, IPSO respectively and memory usage cost minimized by 72% for SJF, 65% for RR, 31.25% for IPSO approaches. In [33], a scheduling model designed by authors which uses deep reinforcement learning approach to effectively schedule tasks coming onto cloud console to Cloud nodes or edge nodes. This scheduling process follows precedence constraints in their tasks, which are incoming to cloud console. It gives a clear distinct mechanism to identify which tasks need to be scheduled to a VM or edge nodes at deployment locations or cache locations of applications. This approach implemented using cloudsim. It compared over several baseline approaches and identified that this approach minimizes 56% of energy consumption, 46% of execution time compared with baseline approaches. In [34], authors focused on

development of a deadline aware scheduling model in fog cloud environment to deal with delicate time sensitive applications. These time sensitive and on demand requirement applications, found more often in IOT environment which may deal smart city applications. These applications changes their behavior according to time and heterogeneity of tasks are also is an important aspect in dealing these kind of applications. Therefore, authors come up with hybridizing MTOA with DQN machine learning model to solve scheduling problem. iFogsim used as a simulation tool for this entire experimentation. It compared over CAG, DNGSA, policy learning approaches. From results, it proved that MTOA-DQN approach shows huge impact over existing policies for makespan, energy consumption. In [35], authors developed a scheduling mechanism, which works with spark jobs in their customized clusters. They developed this customized cluster to check the behavior of spark jobs running in the cluster while maintain SLA objectives. They used DRL based mechanisms for scheduling and workload used by them were real-time AWS instances according to pricing models in Australia. They have used another workload from BigDataBench [36] which consists of heterogeneous jobs i.e. IO sensitive, Network sensitive, Computational sensitive. This entire experimentation conducted on AWS cloud. They Compared this work with existing algorithms i.e. RR, RRC, FF, ILP mechanisms. From experimental results, it proved that DRL based mechanism gain success in minimization of VM cost by 30%. In [37], a computational sensitive based scheduler formulated by authors to effectively schedule tasks among VMs with the use of multi tenancy. A RL based technique used to effectively map tasks to VMs. Simulations carried out on green cloud simulator and evaluated against existing RR, FCFS approaches. From simulation results, it proved that it outperforms existing approaches by minimizing operational costs and maximizing resource utilization. In [38], authors formulated a scheduling algorithm based on RL focuses on improvement of system performance. This algorithm takes heterogeneous requests as input and fed to RL based scheduler to make a decision to schedule tasks in cloud computing. This entire experimentation carried out on Cloudsim. It evaluated against existing algorithms i.e. RR, Max, FIFO, Greedy, Q-Scheduling algorithms. From Simulation results, response time greatly minimized over existing algorithms by 49%, 46%, 44%, 43%, 38% respectively for above mentioned existing algorithms. In [39], authors focused on development of a green fair scheduler in cloud computing which minimizes energy consumption in datacenters. This algorithm uses a DL approach to schedule tasks in this complex system. Simulation carried out on cloudsim. It evaluated against existing conventional migration approach with variable request sizes ranging from 50 to 500 and identified that energy consumption greatly minimized over existing approaches. In [40], authors formulated a scheduling mechanism, which used in edge computing environment. Edge computing suffers from high task failure rate, high service time, high mobility of devices. DRL used as methodology in this algorithm. Edge Cloudsim [41] used as simulation tool for experimentation. It evaluated against existing DQN, PPO approaches. Simulation results revealed that it greatly minimizes service time, task failure rate over DQN, PPO for various heterogeneous workloads. In [42], a multi workflow scheduling mechanism developed for IaaS

clouds to minimize makespan, cost. Multi agent DQN model used for developing this approach, which takes input as multiple workflows with variable number of VMs. Experimentation carried out on real time AWS environment. It compared over existing NSGA-II, MPSO, GTBGA approaches. From simulation results, it proved that multi agent DQN model which takes scheduling decisions based on no prior knowledge outperforms existing algorithms for specified parameters. In [43], Reliability taken as primary objective for design of scheduler in cloud environment. Authors identified a multi agent approach, which takes your task to global queue, and then it will schedule based on buffer capacity and consumed resource usage. For learning purpose, this algorithm uses neural network and it combined with RL approach thereby achieving rewards based on metrics addressed by authors. It implemented on customized simulation environment and it compared against greedy, FIFO, Random approaches. Simulation results shown that makespan minimized to great extent while success rate of tasks, VM utilization rate increased to a good extent. In [44], a dynamic scheduler for cloud environment designed based on Sched RL. This approach transforms existing multi-NUMA scheduler used in existing approaches. Sched RL used 1500 epochs to run entire simulation and gives delta rewards for corresponding parameters i.e. allocation rate, fulfill number of tasks. Authors also mentioned that Sched RL have two limitations i.e. scalability, generalization. It implemented on a real time Azure cloud environment with variable workloads. It compared over First fit, best fit heuristics, and from results, it proved that proposed approach shown huge impact over existing approaches for mentioned parameters. In [45], workflow scheduling formulated for multiple workflows, which designed for prioritizing tasks based on its type and quality of service need to be delivered to customer. This algorithm mainly deals with task ordering into execution mode based on their priorities to get load balance among all nodes. To achieve their goal, authors used RL model, which takes decisions, based on input of tasks, type of tasks and priorities. Simulation carried out on Cloudsim and it compared over Q- learning, Random, Mixed scheduling techniques. Results revealed that RL model outperforms existing approaches by minimizing SLA Violations and maximizing resource utilization. In [46], authors proposed an energy efficient VM scheduling technique, which minimizes energy consumption, SLA violations while maintaining QoS. This work mainly focuses on extracting QoS information from datacenters by making it to learn by using DRL model. It compared with different existing resource allocation mechanisms and extensive simulations conducted on Cloudsim. From simulation results, it shown a huge impact over existing allocation mechanisms for above-mentioned parameters. In [47], a cost based scheduling algorithm formulated by authors to schedule VM instances in Cloud Computing. DRL used as methodology to schedule instances in an effective way. Cloudsim used as simulation tool for simulation. It compared over existing algorithms i.e. Random, RR, Earliest approaches. From simulation results it proved that it shown huge impact over existing algorithms for parameters i.e. Response time, cost, success rate of tasks.

Table 2, clearly shown that many of ML approaches used for resource scheduling algorithms addressed baseline parameters but scheduling in cloud computing

Table 2 Summary of ML approaches for resource scheduling algorithms in Cloud Computing

References	Methodology	Objectives of resource scheduling algorithms modeled by ML approaches
[32]	DRL-LSTM	CPU usage cost, memory usage cost
[33]	DRL	Execution time, energy consumption
[34]	MTOA-DQN	Makespan, energy consumption
[35]	DRL	VM cost
[37]	RL	Operational costs, resource utilization
[38]	RL	Response time
[39]	DL	Energy consumption
[40]	DRL	Service time, task failure rate
[42]	DQN	Makespan, cost
[43]	RL	Makespan, VM utilization rate
[44]	SchedRL	Allocation rate, fulfill number of tasks
[45]	RL	SLA violation, resource utilization
[46]	DRL	Energy consumption, SLA violation, QoS
[47]	DRL	Response time, cost, success rate

environment is highly dynamic and more over that many of researchers used RL and DRL approaches to address problems in resource scheduling. To make decisions more accurate and precisely with heterogeneous workloads. In this chapter, we employed a Deep reinforcement learning approach i.e. DDQN model which is based on RL.

From the extensive literature reviewed in Sect. 2, we identified that many of existing scheduling algorithms formulated based on nature-inspired approaches, which schedules tasks with near optimal solutions. Therefore still there is a challenge exists for researchers to map upcoming dynamic workloads onto suitable virtual resources. Therefore a prominent scheduling approach is needed which should dynamically behaves and allocate requests based on upcoming workloads by considering underlying virtual resources. Therefore, we thought that a machine learning mechanism need to be employed which should consider upcoming workloads and considering underlying resources, which also need to minimize energy consumption, makespan.

3 Double Deep Q-Networks

When using DQN, there is a chance that the Q values will be overestimated, which can result in the underutilization of resources, an increase in the makespan, and the need to wait for the tasks. We use double deep Q networks (DDQN) to get around the problems with the DQN when it comes to scheduling tasks in cloud-based environments.

To begin, we divide the available resources into three distinct categories. The first one is the bandwidth of the network in relation to the links that are established between the switches and routers. Second, the processing power of VMs in terms of CPU and Memory. The third issue is the accessibility of the data in relation to its storage when it is spread out across multiple locations. An environment that contains all three of these types of resources is known as a reinforcement learning environment. We consider Q1 and Q2 to be the queuing models that define the agents, with the Q1 agent being determined by the resources needed to carry out the tasks in the queue and the Q2 agent being determined by the resources that are readily available in the datacenter. The random weights for Q1 and Q2 are the first things that we look at. These values are updated as Algorithm 2 performs its processing of the input. In the system we suggested, we started by setting up the cloud environment and giving each agent its starting weights, as described in Algorithm 1.

Algorithm 1: Configuring Cloud Environment and Setting Up Agent

```

Input: CPU resources, Memory resources network resources,
       storage resources
for i=1 to l //where l is the number of CPU and Memory
resources of VMs in the cloud
   $T_{cm}^i = T_c^i + T_m^i + T_{cm}^{i-1}$ 
for i=1 to n //where n number of available network
links to VMs in the cloud
   $L_b^i = L_b^i + L_b^{i-1}$ 
for i=1 to m //where m number storage components
associated with the VMs in the cloud
   $St_v^i = St_v^i + St_v^{i-1}$ 
//Creating Q1 agent
for i=1 to  $t_k$  //where  $t_k$  is the number of tasks
on queue
   $t_w = w_i$  //initial task weights to
random weights  $w_i$ 
//Creating Q2 agent
for j=1 to  $r_k$  //where  $t_k$  is the number of
available VMs on queue
   $t_w = w_j$  //initial available VM
resource weights to random weights  $r_w$ 

```

After setting up all resource configurations and Q1 and Q2 agents, Algorithm 2 starts executing.

Algorithm 2: DDQL Task Scheduling

```

Input: total number of network resources, vm resources, and
storage rescues, Q1, Q2 agents
Output: updating the Q1, Q2 agents rewards, select action
agent
for i=1 to  $a_j$  //where  $a_j$  number of agents
  task_rewards=0
  for j=0 to n //where n number of tasks
     $Q1(s, ETET_j) = t_{end\_time} - t_{start\_time}$  //where
estimated task execution time

```



```

    Q2(s,AVMRi) = val(Ticm,,, Lib, Stiv)
  If UPDATE(ETETj) then
    Define aj = arg_maxa Q1(s',a) // where a is ETETj
    Q1(s,a) ← Q1(s,a) + α (s,a) (r+ γ Q2(s',aj) - Q1(s,a))
  else If UPDATE(AVMRj) then //where AVMRi
is available VM resources
    Define bj = arg_maxb Q1(s',b) // where a is AVMRi
    Q2(s,b) ← Q2(s,b) + α (s,b) (r+ γ Q1(s',bj) - Q2(s,b))
  S ← s'

```

The initial implementation of the Double Q-learning algorithm makes use of two independent estimates, which are denoted by Q1 and Q2. We use estimate Q1 to determine the action that will maximize profit when the probability is 0.5, but we also use it to update Q1. In Q1 calculations we consider the estimated task execution time, will update the reward of task in Q1, similarly for Q2 calculations we consider the available VM resources, based on the resource utilization of VMs the VM reward are updated. If any update in Q1 or Q2 the state parameters will be updated(s').

By carrying out this procedure, we are able to obtain an unbiased estimator Q1(state, argmax Qnext state, action) for the expected value of Q and inhibit bias.

In our approach we use the present best for selection of the action agent.

$$Q^*(s_t, a_t) = \mathbb{E}_{s'} [R_{t+1} \gamma \max_a Q^*(s', a') | s, a] \quad (1)$$

With \mathbb{E}_s equal to the Q-value of the state-action pair plus the learning rate. The algorithm's reliability on the objective reward value is a function of its learning rate. A discount factor, regulates the relative value of present and future benefits. From Algorithm 2, the time required to select the best agent to execute the task on available resources $O(m*n)$, where m is number of agents and n in number of tasks in cloud environment.

4 Simulation and Results

This section clearly discusses about entire simulation and results of our work. Our simulation carried out on cloudsim [14] simulator which is a discrete event simulator written in Java. It was developed at university of Melbourne. It simulates cloud environment with different policies mentioned by developers. Users can customize and add their policies to evaluate different parameters over this simulator. Therefore, we have chosen this simulator to implement our scheduling model. In this work, simulation carried out by using two different types of workloads. Initially we have fabricated our datasets with different workload distributions and given as input to algorithm and later we are enthusiastic to evaluate efficiency of our approach using a heterogeneous workload real time benchmark dataset mentioned in [36] i.e. BigdataBench which consists of different types of tasks with different heterogeneities. Fabrication of dataset done in 4 types i.e. Uniform distribution-

Table 3 Configuration settings for simulation

Name of the entity	Quantity
No. of tasks	100–1000
Length of tasks	100,000
Computational capacity of physical host	32 GB
Storage capacity of physical host	5 TB
Network bandwidth	1500 mbps
No. of VMs	40
Computational capacity of a VM	2GB
Network bandwidth of a VM	150 mbps
Hypervisor	Xen
Operating system	Linux
No. of Datacenters	8

which consists all types of equally distributed tasks. Normal distribution-which consists of more medium distribution of tasks and less number of small, large tasks. Left Skewed distribution-which consists of more small tasks and less large tasks. Right Skewed distribution-which consists of more large tasks and less small tasks. All these distributions represented as r1, r2, r3, r4 respectively. After fabrication of these dataset distributions. BigdataBench [36] represented as r5. Our proposed DDQN model evaluated against existing RR [48], FCFS [49], EDF [50] algorithms. Table 3 represents configuration settings for our simulation.

5 Evaluation of Makespan

In this section, we clearly presents evaluation of makespan, as it is a primary influential parameter for cloud computing paradigm. This parameter evaluated using above configuration settings mentioned in Table 3. We have given r1, r2, r3, r4, r5 workloads as input to algorithm as mentioned above with different distributions. We evaluated DDQN approach against existing RR, FCFS, EDF algorithms. DDQN run for 50 iterations. Table 4 represents evaluation of makespan for 100 to 1000 tasks. For considered workload r1, when DDQN used makespan generated for 100, 500, 1000 tasks 587.34, 624.99, 1458.37 respectively. For considered workload r2, when DDQN used makespan generated for 100, 500, 1000 tasks 512.89, 945.89, 1034.36 respectively. For considered workload r3, when DDQN used makespan generated for 100, 500, 1000 tasks 543.92, 987.23, 1327.9 respectively. For considered workload r4, when DDQN used makespan generated for 100, 500, 1000 tasks 412.89, 523.78, 866.24 respectively. For considered workload r5, when DDQN used makespan generated for 100, 500, 1000 tasks 769.35, 1023.56, 1356.78 respectively.

Table 4 represents evaluation of makespan of DDQN algorithm over existing algorithms i.e. RR, FCFS, EDF respectively by using fabricated dataset distributions and a real-time benchmark dataset used to test makespan of our approach. From

Table 4 Evaluation of makespan

Algorithms				
No. of tasks	RR	FCFS	EDF	DDQN
r1				
100	793.98	654.76	773.78	587.34
500	967.45	1146.98	876.29	624.99
1000	1562.89	2376.98	2178.34	1458.37
r2				
100	905.76	856.32	798.88	512.89
500	1124.78	1267.90	1078.45	945.89
1000	1892.67	2035.78	1224.23	1034.36
r3				
100	867.23	747.89	623.87	543.92
500	1227.98	1367.65	1164.24	987.23
1000	1562.79	1923.65	1743.87	1327.9
r4				
100	785.73	689.99	534.7	412.89
500	1124.99	1038.78	865.45	523.78
1000	1323.92	1534.78	1425.79	866.24
r5				
100	1867.56	1745.34	1265.23	769.35
500	2034.78	1672.23	1429.9	1023.56
1000	2989.21	3126.78	2578.89	1356.78

Table 4, it is evident that for all distributions and benchmark dataset makespan of DDQN is greatly minimized over existing approaches.

The above Fig. 4 represents evaluation of makespan using DDQN over existing approaches i.e. RR, FCFS, EDF by using various distribution of workloads and real time benchmark dataset i.e. r1, r2, r3, r4, r5 respectively. It is evident that in all the cases our evaluated makespan is outperformed over existing approaches as mentioned in above Fig. 4.

5.1 Evaluation of Energy Consumption

In this section, we clearly presents evaluation of energy consumption, as it is an important parameter for both cloud provider and user. This parameter evaluated using above configuration settings mentioned in Table 3. We have given r1, r2, r3, r4, r5 workloads as input to algorithm as mentioned above with different distributions. We evaluated our proposed DDQN approach against existing RR, FCFS, EDF algorithms. Proposed DDQN run for 50 iterations. For considered workload r1, when DDQN used Energy Consumption generated for 100, 500, 1000 tasks 34.67, 51.45, 89.27 respectively. For considered workload r2, when DDQN used Energy Consumption generated for 100, 500, 1000 tasks 43.24, 59.23, 78.45 respectively.

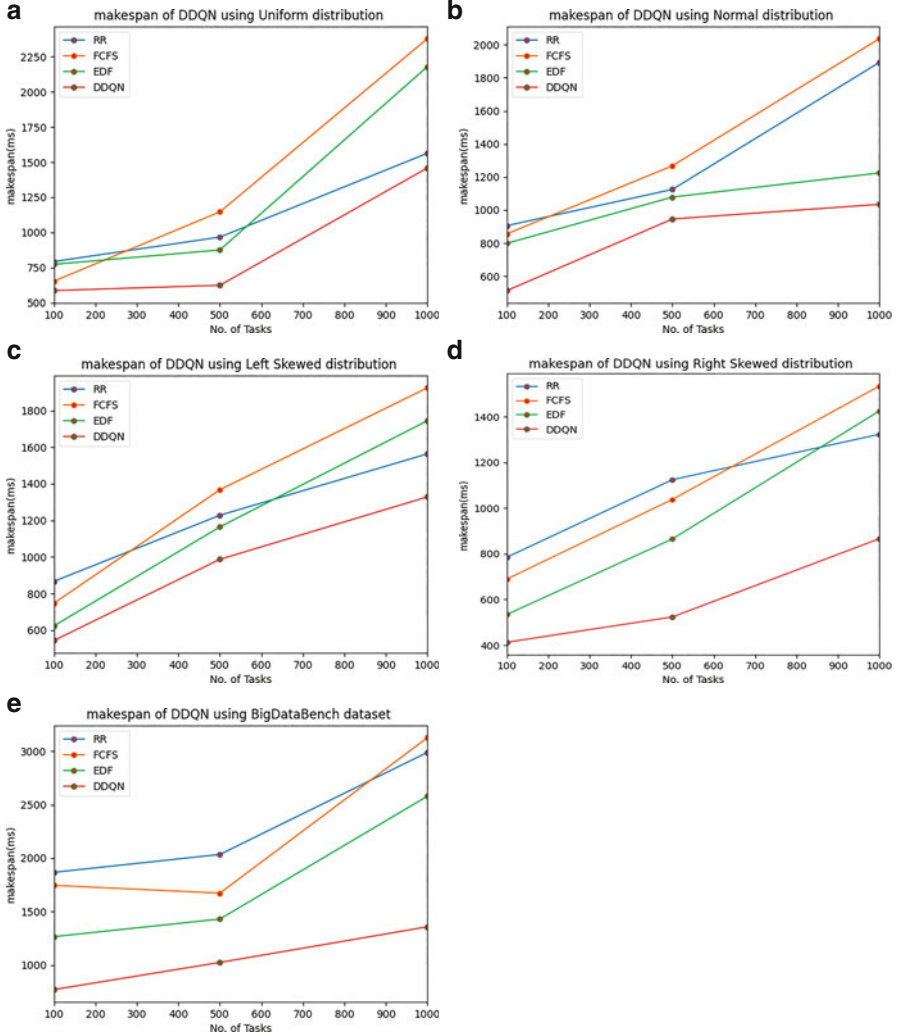


Fig. 4 Evaluation of makespan using DDQN (a) Uniform Distribution of Tasks. (b) Normal Distribution of Tasks. (c) Left Skewed Distribution of Tasks. (d) Right Skewed Distribution of Tasks. (e) BigDataBench worklogs

For considered workload r3, when DDQN used Energy Consumption generated for 100, 500, 1000 tasks 49.23, 56.45, 90.35 respectively. For considered workload r4, when DDQN used Energy Consumption generated for 100, 500, 1000 tasks 38.26, 45.78, 75.38 respectively. For considered workload r5, when DDQN used Energy Consumption generated for 100, 500, 1000 tasks 74.29, 81.56, 90.22 respectively. Table 5 represents evaluation of energy consumption for 100 to 1000 tasks.

Table 5 Evaluation of Energy Consumption

Algorithms				
No. of tasks	RR	FCFS	EDF	DDQN
r1				
100	88.99	94.35	72.86	34.67
500	100.36	106.88	92.45	51.45
1000	143.25	123.99	108.76	89.27
r2				
100	92.67	103.56	84.34	43.24
500	104.65	89.46	92.22	59.23
1000	124.24	135.89	120.56	78.45
r3				
100	87.57	91.45	79.23	49.23
500	93.99	100.56	94.67	56.45
1000	114.2	124.78	105.22	90.35
r4				
100	89.34	92.11	87.56	38.26
500	73.78	98.21	99.14	45.78
1000	103.24	121.67	114.67	75.38
r5				
100	108.56	94.67	98.22	74.29
500	124.79	114.56	106.89	81.56
1000	137.35	121.78	114.26	90.22

Table 5 represents evaluation of energy consumption of DDQN algorithm over existing algorithms i.e. RR, FCFS, EDF respectively by using fabricated dataset distributions and a real-time benchmark dataset used to test energy consumption of our approach. From Table 4, it is evident that for all distributions and benchmark dataset energy consumption of DDQN is greatly minimized over existing approaches.

The above Fig. 5 represents evaluation of Energy Consumption using DDQN over existing approaches i.e. RR, FCFS, EDF by using various distribution of workloads and real time benchmark dataset i.e. r1, r2, r3, r4, r5 respectively. It is evident that in all the cases our evaluated energy consumption is outperformed over existing approaches.

6 Conclusions and Future Research Directions

Resource scheduling in cloud computing paradigm is a huge challenge because incoming requests onto cloud console varies in terms of processing capacities. Therefore scheduling these wide varieties of requests onto virtual resources in cloud is a challenge for cloud provider. Improper mapping of requests to virtual resources leads to decay in system performance i.e. increase in makespan and consumption of energy can be increased which affects both cloud user and provider. Existing

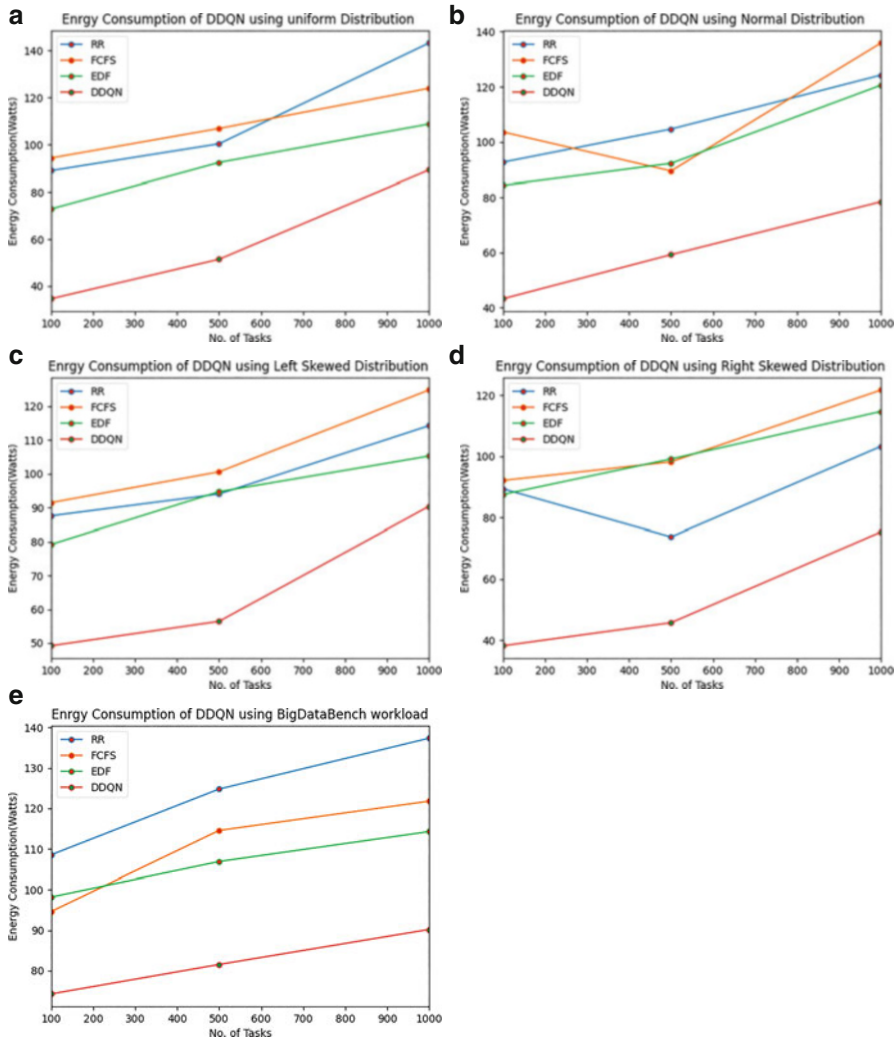


Fig. 5 Evaluation of Energy Consumption using DDQN. (a) Uniform Distribution of Tasks. (b) Normal Distribution of Tasks. (c) Left Skewed Distribution of Tasks. (d) Right Skewed Distribution of Tasks. (e) BigDataBench worklogs

authors used metaheuristic approaches to design schedulers and solve scheduling problems by taking it to near optimal solutions but cloud paradigm is dynamic in terms of requests heterogeneity and to make appropriate decision making out of incoming requests onto virtual resources. In this chapter, we have used DDQN model, which is a reinforcement learning approach fed to the scheduler module helps to take decisions considers task priorities and underlying resource capacity. Entire simulations carried out on cloudsim. Workload for algorithm considered from

a real-time benchmark dataset and datasets fabricated using different distributions. Our proposed approach compared over existing RR, FCFS, EDF algorithms and simulation results revealed that proposed approach using DDQN model shown great impact in makespan, Energy Consumption.

7 Future Research Directions

To manage dynamic workloads, SLA guarantee services, and resource opinions, AI-based algorithms are crucial. However, the various cloud scheduling algorithms, including AI-based algorithms, are constrained by limited(multi-objective) parameters such as task execution time, available resources, etc., to schedule the tasks, which makes the existing solutions considered near-optimal solutions. Future AI algorithms must take into account the following factors in order to achieve optimal solutions in a cloud environment.

Although virtual machines are capable of handling a variety of workloads, the current scheduling algorithms require users to deploy their virtual machines in order to run a single application. If users run many apps, these existing methods deliver average performance. Thus, dynamic workloads on each VM must be taken into account by future AI-based algorithms.

Although virtual and physical resource clustering improves QoS services, the current VM and physical clusters remain essentially static until an auto-scaling event occurs, with overutilization and underutilization of resources as a result. In order to overcome this, AI algorithms must take into account the dynamic clusters in cloud environments, where VMs must cooperate with one another.

Study the static workloads in the cloud that can provide better performance using static schedulers because AI solutions in the cloud environment do not come with free computing and storage.

The risk associated with cloud scheduling algorithms grows with the use of AI-based algorithms. Attackers in this case create dynamic traffic using bots, which can bypass the cloud security measures and result in denial of service attacks. To lower risk in the cloud, AI scheduling algorithms need to be able to find malicious payloads ahead of time.

References

1. Low, C., Chen, Y., & Mingchang, W. (2011). Understanding the determinants of cloud computing adoption. *Industrial Management & Data Systems*, 111(1006).
2. Khallouli, W., & Huang, J. (2021). Cluster resource scheduling in cloud computing: Literature review and research challenges. *The Journal of Supercomputing*, 78, 1–46.
3. Pires, A., Simão, J., & Veiga, L. (2021). Distributed and decentralized orchestration of containers on edge clouds. *Journal of Grid Computing*, 19(3), 1–20.

4. Hustad, E., & Olsen, D. H. (2021). Creating a sustainable digital infrastructure: The role of service-oriented architecture. *Procedia Computer Science*, 181, 597–604.
5. Rashid, A., & Chaturvedi, A. (2019). Cloud computing characteristics and services: A brief review. *International Journal of Computer Sciences and Engineering*, 7(2), 421–426.
6. Diaby, T., & Rad, B. B. (2017). Cloud computing: A review of the concepts and deployment models. *International Journal of Information Technology and Computer Science*, 9(6), 50–58.
7. Tadapaneni, N.R. (2017). *Different types of cloud service models*.
8. Singh, S., & Chana, I. (2016). A survey on resource scheduling in cloud computing: Issues and challenges. *Journal of grid computing*, 14(2), 217–264.
9. Kumar, M., & Sharma, S. C. (2020). PSO-based novel resource scheduling technique to improve QoS parameters in cloud computing. *Neural Computing and Applications*, 32(16), 12103–12126.
10. Ma, J., et al. (2016). A novel dynamic task scheduling algorithm based on improved genetic algorithm in cloud computing. In *Wireless communications, networking and applications* (pp. 829–835). Springer.
11. Pham, N. M., & Nhut, and Van Son Le. (2017). Applying Ant Colony System algorithm in multi-objective resource allocation for virtual services. *Journal of Information and Telecommunication*, 1(4), 319–333.
12. Madni, S. H. H., et al. (2019). Multi-objective-oriented cuckoo search optimization-based resource scheduling algorithm for clouds. *Arabian Journal for Science and Engineering*, 44(4), 3585–3602.
13. Mangalampalli, S., Swain, S. K., & Mangalampalli, V. K. (2022). Multi objective task scheduling in cloud computing using cat swarm optimization algorithm. *Arabian Journal for Science and Engineering*, 47(2), 1821–1830.
14. Calheiros, R. N., et al. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1), 23–50.
15. Strumberger, I., et al. (2019). Resource scheduling in cloud computing based on a hybridized whale optimization algorithm. *Applied Sciences*, 9(22), 4893.
16. Sanaj, M. S., Joe, P. M., & Prathap. (2020). Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere. *Engineering Science and Technology, an International Journal*, 23(4), 891–902.
17. Madni, S. H. H., et al. (2019). Hybrid gradient descent cuckoo search (HGDCS) algorithm for resource scheduling in IaaS cloud computing environment. *Cluster Computing*, 22(1), 301–334.
18. Midya, S., et al. (2018). Multi-objective optimization technique for resource allocation and task scheduling in vehicular cloud architecture: A hybrid adaptive nature inspired approach. *Journal of Network and Computer Applications*, 103, 58–84.
19. Prasanna Kumar, K. R., & Kousalya, K. (2020). Amelioration of task scheduling in cloud computing using crow search algorithm. *Neural Computing and Applications*, 32(10), 5901–5907.
20. Almezeini, N., & Hafez, A. (2017). Task scheduling in cloud computing using lion optimization algorithm. *International Journal of Advanced Computer Science and Applications*, 8, 11.
21. Arul Xavier, V. M., & Annadurai, S. (2019). Chaotic social spider algorithm for load balance aware task scheduling in cloud computing. *Cluster Computing*, 22(1), 287–297.
22. Alsaidy, S. A., Abbood, A. D., & Sahib, M. A. (2020). Heuristic initialization of PSO task scheduling algorithm in cloud computing. *Journal of King Saud University-Computer and Information Sciences*, 34(6), 2370–2382.
23. Srichandan, S., Kumar, T. A., & Bibhudatta, S. (2018). Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm. *Future Computing and Informatics Journal*, 3(2), 210–230.
24. Moon, Y. J., et al. (2017). A slave ants based ant colony optimization algorithm for task scheduling in cloud computing environments. *Human-centric Computing and Information Sciences*, 7(1), 1–10.

25. Shirani, M. R., & Safi-Esfahani, F. (2021). Dynamic scheduling of tasks in cloud computing applying dragonfly algorithm, biogeography-based optimization algorithm and Mexican hat wavelet. *The Journal of Supercomputing*, 77(2), 1214–1272.
26. Awad, N., Mz, A., Liang, J. (2016). *Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective bound constrained real-parameter numerical optimization*. Technical report, Nanyang Technology University, Singapore
27. Shukri, S. E., et al. (2021). Enhanced multi-verse optimizer for task scheduling in cloud computing environments. *Expert Systems with Applications*, 168, 114230.
28. Thanka, M., Roshni, P. U., & Maheswari, and E. Bijolin Edwin. (2019). An improved efficient: Artificial Bee Colony algorithm for security and QoS aware scheduling in cloud computing environment. *Cluster Computing*, 22(5), 10905–10913.
29. Nasr, A. A., et al. (2019). Cost-effective algorithm for workflow scheduling in cloud computing under deadline constraint. *Arabian Journal for Science and Engineering*, 44(4), 3765–3780.
30. Fanian, F., Bardsiri, V. K., & Shokouhifar, M. (2018). A new task scheduling algorithm using firefly and simulated annealing algorithms in cloud computing. *International Journal of Advanced Computer Science and Applications*, 9, 2.
31. Khorsand, R., & Ramezanpour, M. (2020). An energy-efficient task-scheduling algorithm based on a multi-criteria decision-making method in cloud computing. *International Journal of Communication Systems*, 33(9), e4379.
32. Rjoub, G., et al. (2021). Deep and reinforcement learning for automated task scheduling in large-scale cloud computing systems. *Concurrency and Computation: Practice and Experience*, 33(23), e5919.
33. Jayanetti, A., Halgamuge, S., & Buyya, R. (2022). Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments. *Future Generation Computer Systems*, 137, 14–30.
34. Shruthi, G., et al. (2022). Mayfly Taylor optimisation-based scheduling algorithm with deep reinforcement learning for dynamic scheduling in fog-cloud computing. In *Applied computational intelligence and soft computing*. Hindawi Limited.
35. Islam, M. T., Karunasekera, S., & Buyya, R. (2021). Performance and cost-efficient spark job scheduling based on deep reinforcement learning in cloud computing environments. *IEEE Transactions on Parallel and Distributed Systems*, 33(7), 1695–1710.
36. Wang, L., et al. (2014). Bigdatabench: A big data benchmark suite from internet services. In *2014 IEEE 20th international symposium on high performance computer architecture (HPCA)*. IEEE.
37. Suresh Kumar, D., & Jagadeesh Kannan, R. (2020). Reinforcement learning-based controller for adaptive workflow scheduling in multi-tenant cloud computing. *Journal of Electrical Engineering & Education*, 0020720919894199.
38. Mostafavi, S., Fatemeh, A., & Sarram, M. A. (2020). *Reinforcement-learning-based foresighted task scheduling in cloud computing* (pp. 387–401)
39. Karthiban, K., & Raj, J. S. (2020). An efficient green computing fair resource allocation in cloud computing using modified deep reinforcement learning algorithm. *Soft Computing*, 24(19), 14933–14942.
40. Zheng, T., et al. (2022). Deep reinforcement learning-based workload scheduling for edge computing. *Journal of Cloud Computing*, 11(1), 1–13.
41. Sonmez, C., Ozgovde, A., & Ersoy, C. (2018). Edgecloudsim: An environment for performance evaluation of edge computing systems. *Transactions on Emerging Telecommunications Technologies*, 29(11), e3493.
42. Wang, Y., et al. (2019). Multi-objective workflow scheduling with deep-Q-network-based multi-agent reinforcement learning. *IEEE Access*, 7, 39974–39982.
43. Balla, H. A. M., Sheng, C. G., & Jing, W. (2021). Reliability-aware: Task scheduling in cloud computing using multi-agent reinforcement learning algorithm and neural fitted Q. *International Arab Journal of Information Technology*, 18(1), 36–47.
44. Sheng, J., et al. (2022). Learning to schedule multi-NUMA virtual machines via reinforcement learning. *Pattern Recognition*, 121, 108254.

45. Zhong, J. H., et al. (2019). Multi workflow fair scheduling scheme research based on reinforcement learning. *Procedia Computer Science*, 154, 117–123.
46. Wang, B., Liu, F., & Lin, W. (2021). Energy-efficient VM scheduling based on deep reinforcement learning. *Future Generation Computer Systems*, 125, 616–628.
47. Cheng, F., et al. (2022). Cost-aware job scheduling for cloud instances using deep reinforcement learning. *Cluster Computing*, 25(1), 619–631.
48. Alhaidari, F., & Balharith, T. Z. (2021). Enhanced round-robin algorithm in the cloud computing environment for optimal task scheduling. *Computers*, 10(5), 63.
49. Hamid, L., Jadoon, A., & Asghar, H. (2022). Comparative analysis of task level heuristic scheduling algorithms in cloud computing. *The Journal of Supercomputing*, 78, 1–19.
50. Neciu, L.-F., et al. (2021). Efficient real-time earliest deadline first based scheduling for apache spark. In *2021 20th International Symposium on Parallel and Distributed Computing (ISPDC)*. IEEE.

Role of AI for Data Security and Privacy in 5G Healthcare Informatics



Ananya Sheth, Jitendra Bhatia, Harshal Trivedi, and Rutvij Jhaveri

Abstract The encouraging prospects of 5G and Internet of things (IoT) have brought significant advancement in the Healthcare domain. Medical IoT primarily uses cloud computing approaches for real-time remote monitoring of patient's health by employing cyborg-automated techniques such as tele-ultrasound, telestenting and cardiac catheterization. As a result, hospital services have become more convenient and cost-effective. However, the dispersed environment of the sensor-cloud based services poses an enormous threat to patient data privacy and security. Moreover, in a generation dictated by cyber-attacks, data breaches can provide full access to patients' sensitive data such as personally identifiable information and medical history. The necessity to yield new measures for Data Security and Privacy in the epoch of 5G Healthcare Informatics stems from the shortcomings of the prevailing security methodologies like data encryption, third party auditing, data anonymization, etc. In order to address the above challenges and explore the most promising use cases of 5G in the healthcare sector, we discuss the role of Artificial Intelligence (AI), Machine Learning (ML)/Deep Learning (DL) techniques and Blockchain applications that coalesce in overcoming existing hurdles.

A. Sheth

Department of Mathematics, St. Xavier's College, Ahmedabad, Gujarat, India

J. Bhatia (✉)

Department of Computer Science & Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India

e-mail: jitendra.bhatia@nirmauni.ac.in

H. Trivedi

CTO, Softvan Pvt. Ltd., Ahmedabad, Gujarat, India

e-mail: harshal@softvan.in

R. Jhaveri

Department of Computer Science and Engineering, SoT, PDEU, Raysan, Gujarat, India

e-mail: rutvij.jhaveri@sot.pdpu.ac.in

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

M. Kumar et al. (eds.), *6G Enabled Fog Computing in IoT*,

https://doi.org/10.1007/978-3-031-30101-8_2

Keywords Medical Internet of Things · Blockchain · Cloud computing · Privacy and security · Healthcare · Machine learning · Deep learning

1 Introduction

The fourth Industry revolution of the healthcare domain is termed Healthcare 4.0. With the advent of fog computing (FC), Big Data (BD), Medical Internet of Things (MIoT), and 5G technologies, internet health systems have accelerated the global development of emerging innovations. In reality, the digitalization of health and patient data is facing a drastic and transformative change in clinical and economic models. From maintaining manual health records to remote monitoring of patients in real-time, the healthcare industry has seen ground-breaking advancements that have largely benefitted patients as well as doctors. This transition is being fuelled by lifestyle shifts, increased usage of software apps and electronic devices, as well as evidence-based medical treatments rather than irresolute clinical judgments. This results in accurate health diagnosis and improved policymaking protocols. Additionally, these systems aim at eradicating long-standing challenges of remote health, reducing expenses, and enhancing patient care and quality of life (Fig. 1).

IoT-based e-Health systems and sensor-cloud-based computation systems such as fog computing make a promising future trend. Frameworks of IoT-cloud-based e-Health systems vary widely and can be customized to cater to the requirements of specific e-Health system providers. As a result, these providers include a variety of IoT and cloud storage options to allow configuration as per user needs. Research findings of IoT-cloud-based e-Health systems have provided robust security protocols that offer sturdy mechanisms with essential software elements to ensure reliable and safe data transfer between devices. These systems are configured in a way that ensures the evaluation and distribution of data efficiently and without additional delay. Moreover, the deployment of Big data technology is also broadening the effectiveness of ML models for healthcare applications [1]. ML/DL, when paired with IoT and fog computing, can deliver exceptionally precise predictive outcomes. These innovations have the ability to revitalize the healthcare sector, as well as enable online healthcare systems for rural and low-income communities. These innovative establishments have brought about significant changes in the realm of healthcare, however, at the same time, have raised a slew of substantial concerns about the privacy and security of patient data. According to Aksu et al. [2], every three minutes, two computers link to the Internet. As a result of this and the continued expansion of IoT applications, network traffic has escalated severely. Issues such as user data access and protection, as well as application verification and authentication, have emerged owing to increased connectivity. In 2017, 143 million Equifax consumers were compromised of their confidential information [3], thus continuing the upward spike in cyber-attacks. In the same year, a five-billion-dollar toy business had 820,000 customer accounts hacked, according to the report [4]. Cybersecurity disasters continue to prevail in recent history, ranging from

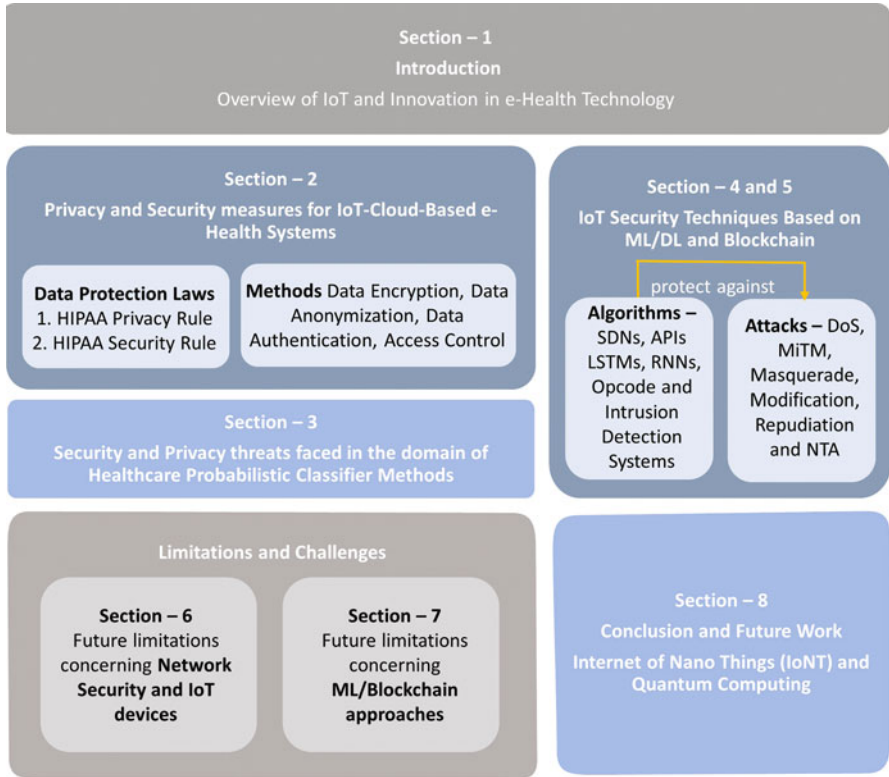


Fig. 1 Structure of the chapter

major data leaks to security vulnerabilities in billions of microchips to operating device lockdowns awaiting payment [5]. Through stringent patient safety measures, seamless data access, remote patient supervision, rapid diagnostic procedures, and decentralized electronic-healthcare records, the implementation of modern Electronic-Health (eHealth) technology technologies will address many issues that plague conventional healthcare systems.

Because of conflicting corporate relationships, the healthcare sector is unable to access and interchange information. As a consequence, there is an unreliable, out-of-date billing mechanism that wastes millions of dollars due to claim discrepancies and conflicts. Due to the deployment of Blockchain, all parties have access to the same information. Blockchain is a decentralized public ledger, which represents a single version of the truth. It enables both stakeholders to supervise and evaluate an underlying asset’s condition in near real-time. Better data sharing among healthcare providers means more reliable diagnoses, more efficient procedures, and an overall improvement in healthcare organizations’ ability to offer cost-effective services. Blockchain technologies will enable different stakeholders in the healthcare value chain to transfer access to their networks without jeopardizing data privacy and

credibility by assisting them in tracing secure data communication as well as any improvements produced. Furthermore, the use of Blockchain (BC) protocols, which are becoming more widespread in IoT applications, protects the underlying healthcare systems from cyber threats. ML algorithms, BC methods, and IoT-based smart have also been the subject of many reports. These projects, on the other hand, use distinct ML, BC, or IoT-based strategies to address security and privacy concerns, demanding a combined survey of recent attempts to address all of the aforementioned challenges utilizing ML algorithms, BC techniques, and IoT devices. In this chapter, we discuss the shortcomings of resource-constrained IoT environments and security concerns in ML/BC techniques, to make healthcare systems and overall e-Health networks more reliable and robust.

1.1 Innovation in e-Health Technology

The pervasive use of IoT systems has boosted innovation and advancement of IoT technology, which now encompasses a variety of architectures for use in health networks. For instance, home monitoring with biosensors has revolutionized the healthcare sector with its ability to treat any chronic disease virtually, by changing the concept of regular hospital visits to frequent monitoring at home. The Health 4.0 Cyber-Physical System (HCPS), in conjunction with Medical IoT, enables patients to receive remotely enabled medicinal treatments via biosensors and actuators. Wearable biosensors, which are instruments connected to a patient's body to measure the presence or concentration of a certain bio-molecule or structure, are used in HCPS. A change in the patient's body state (concentration levels) stimulates signals that are then compared to the patient's medical records, and, based on the protocols, a series of steps are employed to normalize the situation. Finally, bio-actuators are in charge of the remotely triggered release of suitable elements from a capsule inside the patient's body. These medical devices which include bio sensors, glucose monitors, medical wearables, connected inhalers, ingestible sensors, etc. are connected via a networking platform known as the MIoT. It integrates data through the sensors embedded in these devices. Fog Computing is a transitional layer in IoT that provides the interfaces between the sensors and cloud servers thus allowing computing, decision-making, and action-taking to happen via IoT. The kernel of Fog computing is a wireless, low-power, cognitive, powerful computing node that performs analytics and provides insights on raw data collected from various medical devices. The minimization in latency and conserved network bandwidth in FC ensure better analysis and reduced expenses despite of using high computing power.

Moreover, since the COVID-19 epidemic, cyborg-automated tele-ultrasound has captured a significant amount of attention. The merits of ultrasonography over CT include the lack of radionuclides exposure and simplicity of use by qualified ultrasonographers [6]. Recent research using a cyborg-automated tele-ultrasound system involved 23 individuals possessing the virus of COVID-19 (12 with quasi and 11 with acute symptoms), and it was carried out in Wuhan by a professional operating

an MGIUS-R3 system in Hangzhou (Zhejiang, China), some 560 km away from the subjects. The 5G cyborg-automated remote tele-ultrasound framework is still in its primitive stage and has several constraints, including the confinement in terms of patient oriented at the time of inspection in seriously ill patients, poor orientation and curbed functionality of the mechanical robot arm, and use of a solitary ultrasound, according to the reviewers, who also noted that they achieved optimal outcomes in COVID-19 detection.

In addition to that, during cardiac catheterization procedures, stents are implanted using the telestenting technique. Madder and team discovered that standard mechanical angioplasty, in which a humanoid robot manipulator is wired to network controllers in the adjacent chamber, could be carried out at a distance of 55 feet, as part of the REMOTE-PCI trial. In two further studies, Madder used broadband communication to do what he terms “real telestenting” from a mock telestenting lab. In the first trial, the humanoid robot arm was managed from a distance of 4.6 miles, while in the latter, Madder conformed a catheter in a pig from a distance of 103 miles [6].

In addition to demonstrating how drones may serve as airborne base stations (ABSs) to offer broadband connectivity in emergency scenarios, Montero et al. [7] also suggested PERCEPT as a device placement strategy for drones. We think smart hospitals might benefit from an analogous paradigm of thought, where drones can provide connection in minimalistic locations, often completing GBS. As a result, they serve as ABSs. Since distant e-health services may be used with an ABS’ steady and dependable connection, the GBS is no longer inherently obligatory to be the primary driver of internet access. For instance, in contingency planning, cases of emergencies, or even planned circumstances, ABSs are beneficial for teleoperation and remote patient surveillance.

IoT and fog computing are two such breakthroughs that reinforce each other’s abilities. For instance, in their model design, Pasha and Shah [8] focused on connectivity protocols and device criteria for IoT-based smart health systems. Confidence trials revealed that the e-Health system generated by them manifested compatibility and coordination of IoT devices and standard protocols. Their system architecture was also ascertained as stable and secure. Robinson et al. [9] created a framework that integrated smart IoT devices and used cloud storage to provide access to patient data over the internet. It enabled physicians to address their health issues consistently by fostering continuous aggregation of data used to track patient health. The idea of using Fog computing as a transitional layer between the sensor and the cloud server was promulgated by Rahmani et al. [10] as a smart e-Health framework in order to achieve uniform intelligence. Islam et al. [11] employed cloud computing to establish a four-step infrastructure for e-Health systems that included sensors, data aggregation and retrieval, data management, and data interpretation. They outlined the purposes for which the device would be useful, as well as the relevant technology that would be required to make it functional. It also focused on the associated benefits of the technology and its limitations. Shewale et al. [12] presented the idea of an IoT-based body sensor network using compact transceivers with built-in IoT devices for data transmission and reception

through cloud computing. To secure the healthcare system and ensure that patient data remains confidential, they also anticipated privacy and security protocols.

However, IoT-driven healthcare would have to overcome a number of limitations as these devices could be used as gateways for stealing sensitive data if not properly secured. A patient monitoring system that is operating on an outdated version of the software or that is not properly dismantled until after the desired usage, for example, may expose the network to a number of unwarranted cybersecurity risks. Furthermore, the demand for data storage on cloud servers is growing. The study of medical big data using fog computing becomes more difficult as a result of the proliferation of data. Continuous data mining is not only expensive, but it also requires a lot of resources. In addition, Healthcare 4.0 has seen a steady decentralization from the conventional health-centre-based approach, making overall platform management and hardware provisioning less effective. With the introduction of wearables and a slew of modern IoT devices with controlled data flows, improved security is essential and readily available to healthcare professionals. Both of these issues may be addressed with Blockchain technology, which offers interconnectivity, legitimacy, stability, as well as portable user-owned data (Fig. 2).

ML/DL techniques are also widely deployed in the healthcare domain. They benefit four major applications in healthcare: disease prediction, analysis, recovery, and clinical workflow. Patient traits such as phenotypic, genomic, pathology test findings, and diagnostic pictures, are acquired in the clinical environment, which can enable ML models to conduct: disease prediction, analysis, and rehabilitation [14]. For example, machine learning models have been used extensively to identify and classify various types of cancers, such as brain tumours [15] and lung nodules [16]. In healthcare, ML-based approaches are used to extract clinical characteristics in order to speed up the diagnostic process [11].

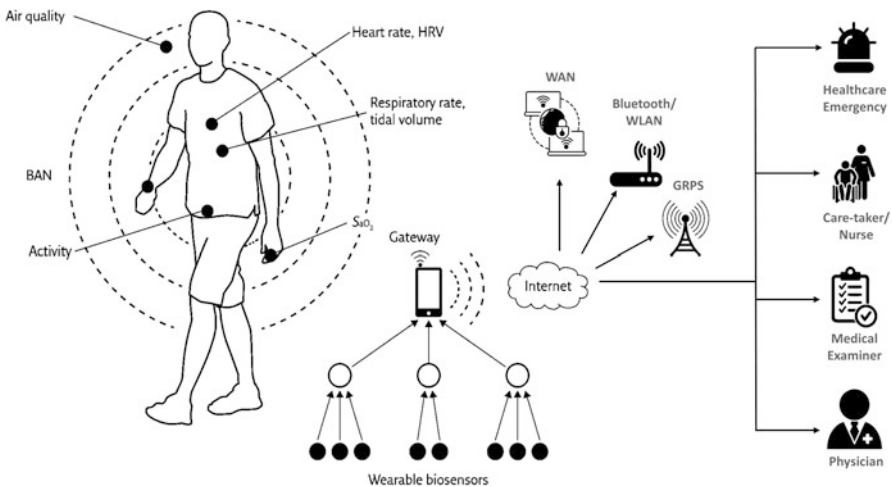


Fig. 2 Role of wearable bio-sensors in the remote monitoring of patients [13]

The pre-processing phase of improving deteriorated medical images has a significant impact on the diagnostic process. As a part of this process, automatic noise is added to the final image many times. Different DL models, such as convolutional de-noising auto-encoders and GANs, have been used to de-noise medical images. Furthermore, scientific NLP advancements are expected to be integrated into potential clinical applications for retrieving useful knowledge from unstructured clinical notes in order to improve clinical practice and testing. Even so, there are a number of roadblocks that prevent ML/DL systems from being used in real-world biomedical systems. We also address the potential risks posed by the use of ML/DL in e-Health applications in this section.

1.2 Real-Time Data Analytics in IoT Using ML

Among the primary problems of IoT systems, one of the most significant challenges is the real-time analysis of big data streams. Analysing IoT data can enable enterprises in delivering high-quality services, make predictions along the lines of new trends, and ensure effective business judgments. IoT data continues to follow the big data pattern in terms of structured framework. Interestingly, with the confluence of ML and AI, the IoT’s potential has significantly increased. Advanced machine intelligence approaches have enabled greater discernment of a variety of real-world situations, as well as the capacity to make essential strategic choices, from the massive flux of IoT sensory data. However, there is a need to build a flexible and adaptable mathematical model with intelligence. In the intelligent healthcare system, these new models may collect data generated by tonnes of IoT-connected nodes. One of the many benefits of an integrated healthcare system is enhanced resiliency and streamlined combination of distinct technologies, lifelong monitoring of patient health, simple accessibility of wearable technology convenience, overall pruning of medical expenditure, and availability of the healthcare practitioners through sophisticated tech-like video conferencing [17] (Fig. 3).

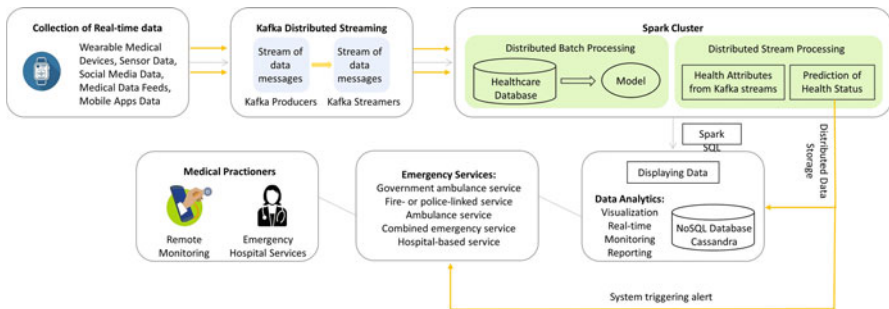


Fig. 3 Real-time data analytics in IoT devices [13]

This section presents a framework for a real-time health prognosis and analytics system utilizing cutting-edge BD techniques. The method relies on using a scalable ML model to analyze streaming health data events imported via Kafka themes into Spark streaming. The procedure is divided into three parts. Initially, Spark is used to convert the conventional decision tree (DT) (C4.5) method into a parallel, dispersed, and fast DT, rather than Hadoop MapReduce, which is restricted to real-time processing. Second, to forecast health status, this approach is used on streaming data acquired from scattered sources of several chronic ailments. At last, the technology assesses health conditions based on multiple input parameters, sends an alarm signal to healthcare practitioners, and stores the information in a distributed database for health data analytics and streaming. Spark DT's is compared to the performance of standard ML technologies like Weka. Finally, model evaluation metrics like execution time and accuracy are computed to demonstrate the technology's efficacy.

2 Measures Undertaken for Practicing Privacy and Security in Healthcare Organizations

2.1 Data Protection Laws

Health insurance includes a wide range of formal and informal data management procedures including health reports, claims data, monitoring, billing records, medical history, and MIoT data. Various organizations such as hospitals, CHCs, doctors, and insurers, use this information to maintain patients' healthcare records. HIPAA (Health Insurance Portability and Accountability Act) was enacted in 1996 to keep track of the actions of medical professionals and other organizational bodies. The act was split into two sections:

HIPAA Privacy Rule – The key goal of this regulation is to protect an individual's health information from unapproved access, disclosure, and exploitation while facilitating the flow of health data necessary to provide and promote better health care and safeguard the health of the general population. HIPAA also includes standards on people's rights to recognize and supervise how their health information is used. The Privacy Rules specify the degree to which an individual's protected health information (PHI) can be revealed and used by institutions subject to these rules. These persons and organizations are referred to as "covered entities," and they include the following:

1. Healthcare providers: These include all healthcare providers who electronically transmit patient records along with certain purchases, regardless of the extent of their practice. Claims, compensation eligibility questions, referral permission appeals, and so forth are examples of these transactions.
2. Health plans: They are organizations that offer or pay for medical benefits. Life insurance, such as those that cover health, medical, dental vision,

prescription drugs, as well as health maintenance organizations (HMOs) and Medicare replacement providers, are all examples of health policies.

3. Healthcare clearinghouses: These consist of entities that transfigure unstructured data received from another organization into a standard tabulated format in order to keep a log of health records. Individually identified Protected Health Information (PHI) is typically only received from healthcare clearinghouses while they are acting as a corporate partner with a health plan or healthcare provider.
4. Business associates: It refers to a person or agency (other than a protected entity’s workforce) who uses or discloses personally identified health information to spread ads and incentivize drug promotions.

HIPAA Security Rule – The HIPAA Security Rule covers a subset of the information protected by the Privacy Rule, while the Privacy Rule protects PHI. This subset includes any electronically generated, obtained, controlled, or distributed personally identified health information by a person. In accordance with the HIPAA Security Rule, all covering agencies must:

- Guarantee the protection, credibility, and availability of all e-PHI
- Identify and safeguard against anticipated risks to the information’s security – Secure against anticipated impermissible uses or disclosures (Fig. 4).

HIPAA, on the other hand, was not strictly enforced; the probability of an audit was minimal, the fines for PHI violations were moderate, and compliance was essentially non-existent. The policies became tougher following the introduction of the HITECH (Health Information Technology for Economic and Clinical Health) act in 2009. In response to what HIPAA suggested, it improved the law’s enforcement by raising fines for noncompliance and forcing company partners to obey the same laws as all protected institutions. As a result, investigations of covered institutions or business partners who violated ePHI (electronic Protected Health Information) or were reported to be violating HIPAA wilfully were conducted periodically. HITECH has urged healthcare providers to use electronic health records, encouraging health information technologies and enhancing healthcare quality, protection, and effec-

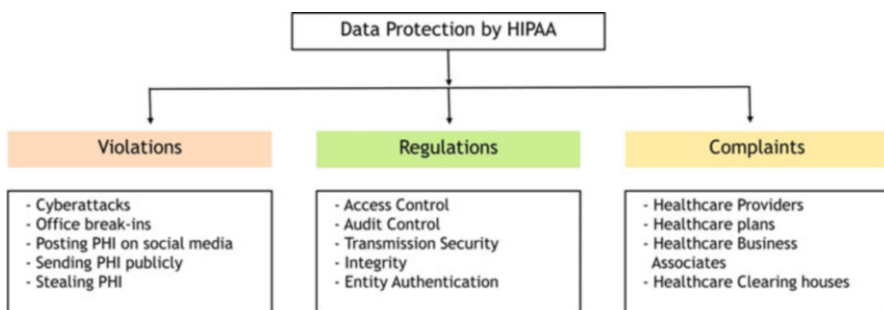


Fig. 4 Data protection under the HIPAA rule

tiveness. The act has proved to be more successful in improving the safety and protection of electronic health records leading to stricter sanctions.

3 Privacy and Security Methodologies Currently Practiced for IoT-Cloud-Based e-Health Systems

Since MIoT devices lack adequate processing power, energy, and computational space, real-time data analysis necessitates efficient and scalable high-performance computing and a vast storage framework. The collected medical data is actually processed in the cloud for the majority of MIoT establishments. Because of their elasticity, cloud providers will leverage collective tools and facilities, enabling a promising approach for efficient healthcare data management. It becomes important to gather data from trusted sources, in order to avoid data disclosure.

The challenge of developing protected information systems is primarily driven by three basic and competing factors: (i) the security technology's complexity; (ii) the complexity in classifying the information to be covered versus the information to be allowed access to; and, ultimately, (iii) human technology use that is right and optimal. Since it deals with "human-system relations," the last aspect is typically the most complicated. Information leakage from the system can lead to disparities in a number of forms, such as being refused health care or jobs based on sickness or genetic information, if adequate privacy and protection precautions are not taken. The current techniques employed to practice privacy and security of data collected through these sources involve:

3.1 Data Encryption

Data encryption is a powerful method for avoiding unauthorized access to confidential information. Even as the data captured is stored in cloud-based storage, the technologies protect and retain data control throughout its lifecycle. Encryption is beneficial in preventing data loss and avoiding vulnerability to breaches. In the process of Data Encryption, the original message, also known as plaintext, is encrypted into cipher text by various encryption techniques (e.g. Homomorphic encryption, garbled circuits, secret sharing, etc.). The message is then transmitted via a public channel from the sender to the recipient. This message is decrypted into plaintext until it reaches the recipient. Patient records must be secured under HIPAA. One of the most common defenses against unauthorized access to a person's sensitive information is data encryption. 4 shows a common model of Data Encryption used to preserve data confidentiality (Fig. 5).

Medical institutions and suppliers must ensure that their encryption system is both effective and competent, as well as user-friendly for both patients and



Fig. 5 Procedure for key exchange

healthcare professionals. It also needs to be capable of adapting to modern electronic health reports. Complex encryption algorithms or communication protocols may have a significant impact on data transmission rates, and in some situations, data transmission can also fail. It is difficult to find a balance between data security and system energy consumption as a result of this. Despite this, field experts have identified a few approaches that use the least number of resources. For instance, a lightweight end-to-end key management scheme, that ensures key exchange with minimal resource consumption was put forward by Abdmeziem and Tandjaoui. It is built on a heterogeneous network that combines nodes with various capacities, resulting in robust security features despite resource constraints. However, as hacking techniques evolve, so should our encryption strategies. To ensure the secure protection of medical records, appropriate encryption algorithms must be applied.

3.2 Data Anonymization

Although it can appear to be analogous to encrypting data, the masked value does not retrieve the original value. One of the most important advantages of this method is that it significantly reduces the expense of securing big data implementation. Masking eliminates the need for extra authentication features to be applied to data when it is in the platform when it is transmitted from a safe source. Explicit identifiers employ a technique of labeling data sets using personal identifiers such as an ID number, a name, a social security number, or a cell phone number while eliminating or generalizing quasi-identifiers such as dates of birth and zip codes. The confidential characteristics of a patient, such as medical status and salary, are referred to as privacy records. When considering the distribution features of the original data, it is important to ensure that the individual properties of the current dataset are adequately processed during the data publishing process, in order to protect the patient’s privacy and identity. Else, patient data might get swapped in the process, leading to a huge ruckus in the system.

At present, data anonymous technology such as k-anonymity, l-diversity, and confidence bounding are typically used to address issues of this sort. In particular, conventional k-anonymity is widely used for the purpose of data anonymization. Even though it prevents patients from identity exposure, it does not shield them from attribute revelation so intruders may use accuracy attacks and context information

to recognize confidential details and personal contacts, resulting in data leakage. The authors also proposed p -sensitive anonymity, a thought-provoking principle that protects from both identification and attributes revelation.

3.3 Data Authentication

Authentication is the process of verifying whether or not statements made by or about a person are accurate and authentic. It serves critical functions in every enterprise to ensure data security, including maintaining access to organizational networks, protecting user identity, and ensuring that the user is who he claims to be. The role of authentication is often undervalued, especially in the fields of medical research and healthcare. For example, in the most recent medical study using online tools, the highest degree of authentication uses only passwords. As a consequence, the information is likely to be exploited. To access the online facilities, the patient must be accredited. Authentication should be required both during the approval process and when signing in after approval and registration. Because if a registered user does not opt-out of the system, other users will log in as that person and access all of the information that was previously submitted. Many healthcare institutions provide their patient's confidentiality security, shared verification, and automated identification. The suggested scheme's main components are patient-owned Near Field Communication (NFC) wristbands and handheld devices issued to clinical professionals. The data used in the process is managed by an intermediate computer. It is then encrypted by a second server with a secret key generator in order to ensure some form of communication. A keyed-hash message authentication code is used to validate the identity of the patient. The amalgamation of the above tools creates a safe and stable mobile health (m-Health) service for managing patient data during emergency and hospitalization services.

However, data authentication does not come without limitations and problems, especially with man-in-the-middle (MITM) attacks. MITM attacks can be classified into two: Spoofing and impersonation. In order to avoid MITM attacks, most cryptographic techniques employ some form of endpoint authentication. Cryptographic approaches such as Transport Layer Protection (TLS) and Secure Sockets Layer (SSL), for example, provide security for communications over networks like the Internet. They encrypt network connections' transport layer segments from beginning to end. Online applications such as web searching, electronic mail, Internet faxing, text calling, and voice-over-IP (VoIP) use these technologies. To achieve authentication, hashing techniques such as SHA-256 (commonly used in bitcoin ledgers) can be used. In addition, the Bull Eye algorithm can be used to keep track of all sensitive data across the network. This algorithm protects data and maintains relationships between real and virtual data. Sensitive data can only be read or written by an authorized user. In a healthcare system, all benefactor-provided healthcare records and customers' identities should be checked at the point of any entry, which takes us to our next approach for data protection and privacy.

3.4 Access Control

Access control – After performing user authentication, users can enter a management system but their access is limited by an access management agreement, which is normally based on the freedoms and immunities of each physician approved by the consumer or a third-party provider.

This is crucial in preventing unwarranted access to facilities. It ensures that the person who has access to the data can only do the tasks for which they have been granted permission. The real challenge emerges when data must be guarded against unauthorized disclosure while being available for inspection or law enforcement purposes. It must also be easy for an authorized user to gain access to the data while challenging for an unauthorized user to execute the same tasks. Over the years, many methodologies have been developed to tackle the issues faced in the domain of access control.

The most commonly used models for the purpose of access control are role-based access control (RBAC) and attribute-based access control (ABAC). However, when used alone in the medical system RBAC and ABAC have demonstrated certain limitations. While noncryptographic approaches lack a safe and consistent framework for implementing access policies, cryptographic approaches are too costly, cumbersome, and restricted in terms of policy specification. However, using cloud-based HIE (Health Information Exchange) technologies, numerous strategies have been created to overcome the problem by harnessing the best of both. The HIE cloud, healthcare organizations (HCOs), and patients are the three primary components of the system (Table 1).

Other alternatives to access control include Identity Based Access Control (IBAC), which grants privileges to individual users based on their requirements, and Mandatory Access Control (MAC), which establishes mandatory rules for all system users. In order to meet the more varying needs of an agency, a model may be hybrid and comprise of more than one model. Since authentication and access control are intertwined, both systems must operate accurately and reliably.

Table 1 Machine learning approaches used for maintaining security in IoT networks

Cyberattacks	ML techniques	Algorithms	Security techniques
DoS	Supervised and unsupervised ML	NN multivariate	Secure IoT
		Correlation	Offloading
		Analysis	Access control
Spoofing	Supervised ML	SVM	Authentication
Jamming	Reinforcement learning	Q- learning, DQN	Secure IoT
			Offloading
Masquerade	Supervised ML	SVM, Naïve Bayes, K-NN, NN	Access control
Malware	Reinforcement learning	Q- learning, dyna-Q,	Malware detection,
	Supervised ML	K-NN	Access control

When protocols become more complicated, providing access control becomes more difficult. These regulations must be closely examined in the healthcare setting so that appropriate access management can be established and implemented without deterring the system's use.

3.5 Trusted Third Party Auditing

Security monitoring, also known as Trusted third-party auditing (TTPA), is the method of collecting and reviewing network activity in order to detect any possible data invasion. The term "audit" refers to the process of tracking user activities in the healthcare system in chronological order, such as keeping a list of every user who accessed the system and/or altered data. The healthcare system will monitor and guarantee its protection using these two optional security metrics. Many times, cloud servers cannot be fully trusted. If the data is corrupted in some way, it loses all of its credibility and utility. Data corruption, for example, can occur at any point in a process, from the host to the storage medium. When data arriving in the system's memory is corrupt, it creates significant ramifications. Without the user's consent, this may also result in data loss. The data rules are typically defined by the user for security purposes so that the service provider cannot alter the source data. Patient records and coding audits are performed by third-party organizations, and the reports are submitted to a government-mandated agency such as the Department of Health and Human Services (HHS). It is also known for providing impartial auditing reports in order to ensure cloud service providers' transparency and secure the legal benefits of data owners. Many auditing approaches have been developed in recent years. To detect suspicious usage, supervised machine learning methods such as logistic regression, support vector machine, decision trees algorithm, and others have been used.

3.6 Data Search

Sensitive data which must be encrypted before being outsourced to preserve data protection renders the standard data use which is dependent on plaintext keyword search completely impractical. As a result, providing an encrypted cloud data search service is critical [18]. Searchable symmetric encryption (SSE) and public-key encryption with keyword search are the two most popular approaches for searchable encryption. It should also be remembered that the more complicated the security methods are, the more difficult it is to scan the data and verify the accuracy of the search results. If the search results cannot be executed in a timely fashion, all security and privacy protections are made useless. To strictly adhere

to privacy protocols, Bezawada et al. [19] established a synchronous key-based solution. To attain powerful query processing functionality over encrypted files, they implemented string matching in cloud servers. The Pattern Aware Secure Search (PASS) Tree which does not disclose any semantic similarity of the keywords, was established as an effective and accurate indexing structure. They also defined a relevance rating algorithm that uses the pattern query to return the most important documents to the user. Experiments on massive real-world data sets containing up to a hundred thousand keywords demonstrated that the suggested algorithm would find patterns in under just several milliseconds with 100% accuracy.

4 Security and Privacy Threats Faced in the Domain of Healthcare

4.1 Leniency in the HIPAA Regulations

For years, experts have been concerned about the moral repercussions of using AI in healthcare data privacy and protection policies. HIPAA has been made obsolete by developments in artificial intelligence. HIPAA is obligated to protect patient health data only when it is provided by healthcare providers, healthcare clearinghouses, and insurance companies. And, since healthcare data is so important to AI businesses, bending a few privacy and ethical principles have become common protocol. Since social media sites don't fall under HIPAA jurisdiction, they may, for example, collect phase data from a user's mobile app without their permission, then purchase health care data from another entity and conflate the two values. They will now have access to a patient's healthcare records that are connected to their identities. They could now either start promoting ads focused on that or sell the data to other organizations such as genetics testing agencies, biotech companies, pharmaceutical companies, and insurance firms, among others. These companies could bias the selection and pricing of healthcare products, alter insurance claims and conduct drug promotions. The Covid-19 pandemic has only aggravated the problem. Penalties for HIPAA violations against healthcare professionals who assist patients via e-communication mediums during the pandemic have been waived under the Public Health Emergency (PHE) protocol, leaving confidential patient records more fragile than ever. Furthermore, sanctions for violations of any other HIPAA Privacy Law protocols against health care providers or company partners for the good faith use and dissemination of PHI for public health purposes are no longer mandatory.

4.2 Resource Constraints for IoT Devices

Because of the available functionalities of end devices, IoT threats vary from traditional network threats [20]. The Internet of Things has little memory and computing capacity, while the traditional Internet has efficient servers and machines with plenty of resources. As a result, multifactor security layers and dynamic protocols can be used to protect a conventional network, which is something that a real-time IoT device cannot do. IoT systems, unlike conventional networks, utilize less secure wireless networking media. Finally, IoT devices have varying data contents and formats due to application-specific features and the absence of a standardized operating system, making it difficult to create a consistent security protocol [21]. Both of these flaws make IoT vulnerable to a variety of protection and privacy breaches, allowing for a variety of attacks. The likelihood of a network attack rises as the scale of the network grows. As a result, the IoT network is more vulnerable than a conventional network. Furthermore, IoT systems that communicate with one another are usually multi-vendor devices that use various benchmarks and guidelines. Communication between such devices is difficult, necessitating the use of a trustworthy third party as a bridge [22]. Furthermore, multiple surveys have raised concerns about the need for billions of smart devices to receive daily app upgrades.

According to Milosevic et al. [23], computationally advanced machines, such as laptop computers, could be able to detect ransomware using advanced resources. IoT systems, on the other hand, have insufficient resources. Standard cybersecurity programs and applications, however, are inefficient at identifying minor threat variants or zero-day attacks [22], since both must be revised on a frequent basis. Furthermore, the provider would not have real-time alerts, leaving the network insecure.

4.3 Active Attacks in Information Security

An active attack tries to change the system's capabilities or functionalities. Active attacks include tampering with the data stream or fabricating misleading statements. The following are examples of active attacks:

1. **Masquerade Attacks** – A masquerade attack occurs as one person assumes the identity of another. These types of attacks are used to obtain unauthorized access to personal computer data through legitimate access verification. Masquerades are often discovered automatically by identifying major deviations from standard user activity. If a user's standard profile differs from their original actions, it could indicate that a masquerade attack is underway. Using stolen login credentials, detecting security holes in applications, or sidestepping the authentication process may all be used in a masquerade attack. The attempt may come from inside an organization, such as from an employee, or from outside the

organization through a public network link. Poor security makes it much simpler for an attacker to obtain access, making it one of the simplest points of entry for a masquerade. Once the intruder has been granted access, they will have direct access to the institution's sensitive data, as well as the ability to alter and erase software and data (contingent on the privilege level they claim to have). Each of the other types of active attacks is used in a Masquerade attack.

2. Trojan Horse Attacks – A hidden virus gives unauthorized intruders entrée to the computer system, connection, or programming software. As an illustration, the hackers could bury malicious spyware under a URL that is really apparent. A trojan will be injected into the gadget after the consumers click the hyperlink. The perpetrators might then have clear control over the system. As a consequence, a kernel is just one more illustration of a Trojan Attack. Typically, a rootkit won't be able to get unauthorized access to a system. The intruders will get admin rights as a result. Threat actors can commandeer the system, but individuals won't be made aware of it.
3. Modification of messages – This refers to the alteration of a part of a message, as well as the delay or reordering of the message, in order to achieve an unintended result. A message that says "Allow Max to read confidential file X" is changed to "Allow Clark to read confidential file Y." Such attacks can be classified as credibility attacks, but they may also be called availability attacks. We also harmed the credibility of the data stored in a file if we gain unauthorized access to it and change the data it holds. However, if the file in question is a client program that controls how a certain service operates, such as a Web server, modifying the contents of the file can have an impact on the service's availability. If we stick with this idea and suggest the setup, we changed in our Web server's file affects how the server handles encrypted connections, we might also claim this as a confidentiality attack.
4. SYN Flooding – The intruders can continue to produce data packets and send them to all or specific client channels. Usually, fake IP and URLs are deployed. The SYN-ACK protocols might then be responded to by the client that is not informed of the intrusion attempt. The system might malfunction if it is unable to contact the users. For cyberattacks like SYN flood, standard statistical strategies may be more likely to produce attack monitoring systems. One such method is provided by writers wherever they are needed, supporting the Bayesian calculator with an SYN flood intrusion identification component for unanticipated cellular operators.
5. Repudiation – Repudiation is an attack that may be carried out by either the sender or the recipient. Later, the sender or recipient may deny having sent or received a packet. For instance, a consumer might ask his bank to "transfer a sum to another account," only to have the sender (customer) reject making such a request later. This effectively ensures that a user may clearly dispute awareness of a transaction or communication and then later argue that the transaction or information exchange never occurred. These kinds of information security attacks usually occur when a program or device fails to implement controls to correctly detect and record users' activities, allowing malicious modification or

falsification of new actions to occur. This attack can be used to alter the details of fraudulent user actions, thus using it to log incorrect data to log files. In a similar way to tampering with mail messages, its use can be applied to general data theft in the interest of others. If this attack succeeds, the information contained in log files will be deemed invalid or deceptive.

6. Denial of Service attacks – Of all the security threats, Denial of Service (DoS) has the most explicit execution. A DoS attack tries to put a device or network to a standstill, making it inaccessible to its targeted consumers. DoS attacks operate by crippling a system with traffic, sending it information that causes it to malfunction, and draining network resources such as latency. In both scenarios, the DoS attack depletes legitimate consumers (employees, operators, or account holders) of the service or resource they sought. As a result, legitimate customers are unable to use these facilities. Distributed DoS (DDoS) is a more sophisticated variant of the DoS attack in which multiple sources attack a single target, rendering the attack more difficult to locate and prevent. DDoS attacks come in a variety of forms, but they all have the same goal. SYN flooding [24] (sends a connection requisition to a server without actually performing the handshake.) Internet Control Message Protocol (ICMP) attacks [25] (persists until all available channels are filled with requisitions and no legitimate consumers are allowed to interact with them), (It exploits the poorly configured network machines by transmitting fraudulent packages that attack the said device on the network rather than a specific device.) and crossfire attacks [26] (in which an attacker uses a complex and ludicrously large botnet for attack execution.) are a few examples of DDoS attacks (Fig. 6).

4.4 Passive Attacks in Information Security

Passive attacks aim to learn or use information from the system without causing any damage to the system's infrastructure. Eavesdropping or control of transmission is the essence of passive attacks. The adversary's aim is to collect knowledge that is being exchanged. The following are examples of passive attacks:

1. Message content distribution – Relevant or confidential material can be included in a telephone call, an electronic mail address, or a transferred file. We want to keep the contents of these transmissions hidden from an adversary. These types of attacks open the network to unwarranted privacy threats and lead to the leakage of sensitive patient data.
2. Eavesdropping attack – An eavesdropping attack occurs when intruders put a malware toolkit within the communication system to record future analysis of the bandwidth utilization. This is due to the escalating connectivity of vulnerable devices, such as those of the IoT ecosystem. To intercept the internet traffic, the intruders must be forced into the access link that connects the end node and the UC system. The perpetrator will find it easier to install a software toolkit into the

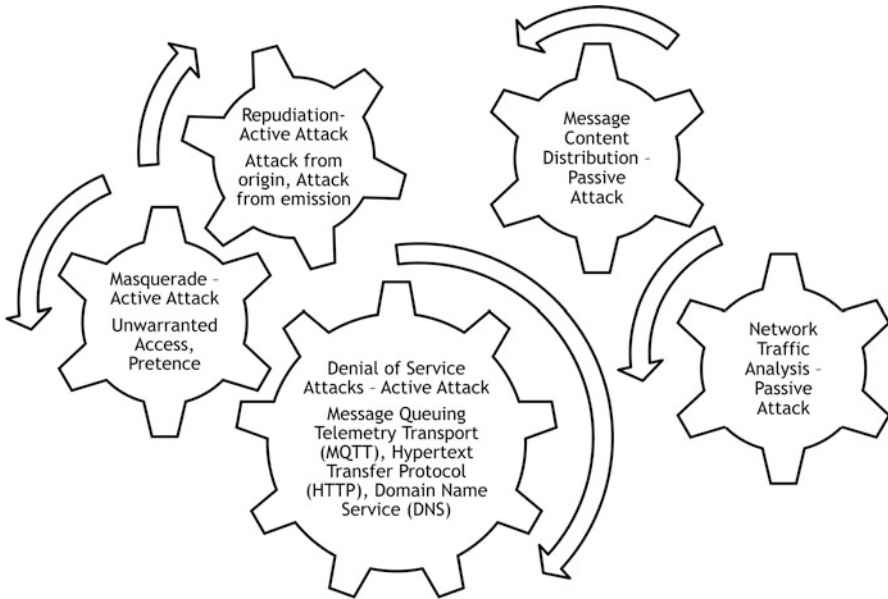


Fig. 6 Active and passive attacks in information security

access link if there are more networking procedures and if the aforementioned techniques are lengthier.

3. **Network Traffic Analysis** – Suppose we had a way of masking (encrypting) information so that even though the message was intercepted, the intruder would be unable to retrieve any information from it. The adversary may ascertain the communication host’s location and identity, as well as the frequency and duration of messages exchanged. This knowledge might be helpful in figuring out what kind of contact was going on. Attackers are constantly changing their methods in order to prevent detection, and they often use authentic credentials with reputable software already installed in a network system, making it impossible for companies to spot critical security threats in advance. In addition to attackers’ incessant creativity, network traffic analysis products have evolved, providing companies with a practical way forward in combating innovative attackers. Furthermore, with the widespread acceptance of cloud storage, DevOps processes, and the Internet of Things, retaining efficient network visibility has become a dynamic and daunting task.

4.5 IoT Devices with Malware

The term “malware” refers to malicious applications. The number of IoT devices has been increasing in recent years, as has the frequency of IoT security updates, which

can be used by an attacker to steal information and bypass spam on a computer to execute malicious activities. Viruses, spyware, worms, Trojan horses, rootkits, and malvertising are all examples of malware. Smart home systems, medical devices, and vehicle alarms are only a few examples of what can be hacked. Malware on IoT was investigated by Azmoodeh et al. [27]. These types of perpetrators are normally state-sponsored, well-funded, and well-trained. Using various supervised ML algorithms, the authors in [28–30] attempted to protect resource-constrained android smartphones from malware threats by employing several supervised ML techniques. A thorough review of malware diagnosis was also presented in which many security flaws in the Android operating system were highlighted, particularly on the application layer, which contains frameworks with a variety of components. However, we shall discuss potential solutions in order to avoid further exploitation of technology. By integrating adequate technical safeguards into their AI healthcare solutions, technology firms will potentially be able to reduce the abuse of those resources. Doing so might reduce the fraction of data breaches in the system architecture. And as a measure to devise such safeguards, discussed below are the most promising AI and ML approaches that balance the potential for positive health outcomes with concerns over health data privacy.

5 Secure, Private and Robust ML/DL Techniques as a Countermeasure Against IoT Threats

In virtually every area of healthcare, machine learning algorithms are used to deter data breaches and protect confidential patient data. ML entails the continuous and meticulous training of algorithms on experimental data in order to develop and improve them. As a result of improved data acquisition techniques and a surplus of medical data to train models, the ML methodology in the healthcare domain has progressed quickly. The foremost advancements to e-Health systems made possible by integrating comprehensive IoT-based smart health strategies are outlined in this section. We also examine current potential security and privacy options for IoT-cloud-based e-Health systems and include essential framework specifications.

5.1 Privacy Isolation Algorithm

An IoT-enabled medical framework for user confidentiality employing big data analytics is meant to distinguish between sensitive data (user’s personally identifiable information) and the corresponding user’s health-related data [31]. In particular, we leverage this approach to segregate the private data since time series data from

smart technology may be transcoded with other behavioural variables leading to the disclosure of a user’s identity. This strategy has both a cloud node and a user node. We differentiate the strolling and relaxing states at the user node and segregate gait data from the privacy-isolation zone. Additionally, the gravity’s impact on these measurements is removed. The processed information is then uploaded to the cloud node without the user’s private details, retaining and maintaining user privacy.

5.2 Software Defined Networks (SDNs)

Software-driven configurable machines (SDNs) are used in combination with Machine Learning, which customizes models based on the user’s needs. Restuccia et al. [32] proposed a taxonomy of current IoT security risks and their responses, which they achieved by combining SDN and ML. They also recommended that the mechanism of collecting data be broken because the primary role of an IoT system is to gather data from IoT instruments such as medical wearables, bio-sensors, glucose meters, and so on. The process was split into three stages to prevent security attacks: IoT authentication, IoT powerline communications, and IoT data aggregation. For example, a Bayesian learning model was used to identify cross-layer malicious attacks, and an ANN prediction model was used to determine the data set’s legitimacy (Fig. 7).

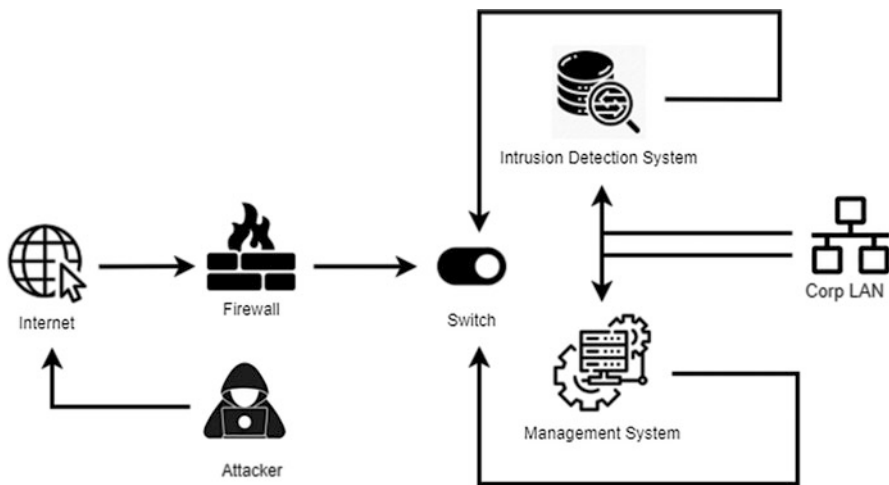


Fig. 7 Network intrusion detection systems

5.3 Network Intrusion Detection Systems (NIDS)

Chaabouni et al. [33] concentrated on IoT-based NIDS that might look for a security breach in a grid. An overview of all potential layer-wise attacks was provided in the IoT-based architecture. The perception layer, network layer, and device layer were the three layers that were classified. They also compared the architecture, detection strategies, and empirical setup of common IoT NIDS. The study focused primarily on machine learning algorithms, but it also demonstrated how Network IDS in IoT could alleviate the difficulties that traditional IoT systems face.

5.4 Application Programming Interfaces (APIs)

Sharmeen et al. [34], proposed simplified API applications for Web developers in which an ML model could be trained using three sorts of attributes: static, dynamic, and hybrid. To conduct a thorough review of each function group, the dataset output measurements, image segmentation process, usability, and security model are all used. They also came to the conclusion that through hybrid analysis, they were able to choose both static and dynamic features to increase computational performance. This paper, however, is confined to a sole security hazard. In contrast to dynamic analysis, [30], the developers of [28] used static analysis methods for feature selection, taking into account all heretofore unstudied Application Platform Interfaces (API). The commonly used features from previous researchers were used to guide feature extraction. For the second largest malware prototype dataset ever used, they achieved a 98.9% accuracy rate. When intrusion methods grew more advanced, static analysis became obsolete, necessitating the use of a more intricate scheme. Attackers used deformation technology to get around static analysis techniques, which allowed them to escape scrutiny, while dynamic analysis approaches looked impressive due to their insusceptibility to code transformation strategies. Based on these questions, the developers of suggested EnDroid, a new paradigm [29]. The suggested approach employed Chi-Square for feature extraction, an ensemble of various models (decision tree, linear support vector machine, highly randomized trees, random forest, and boosted trees) for base classification, and LR as a meta-classifier, yielding a 98.2% precision.

5.5 Deep Learning Models

By implementing the DL model LSTM [35] a significant decrease in the MiTM attacks was observed. The DL model LSTM took somewhat longer to learn than the LR model, but it was 9% more accurate. As LSTM was compared to softmax for multi-class grouping, the resulting precision was 14% higher. Abeshu

and Chilamkurti pointed out in a related analysis that an IoT device's resource limitations make it a possible target for DoS attacks [36]. In a massively distributed network like the Internet of Things, traditional machine learning algorithms are less reliable and scalable for detecting cyber-attacks. Since there is so much data generated by millions of IoT computers, deep learning models can learn faster than superficial algorithms. However, their research centered on decentralized DL for fog computing frameworks using constraints and model sharing. IoT devices' processing capacity and storage space are reduced thanks to fog computing. As a result, it is an ideal spot for spotting an intruder. The current Stochastic Gradient Descent (SGD) for fog-to-things processing requires parallel computation. The centralized SGD would suffocate as a result of the massive volume of data generated by IoT. As a result, the study proposed a dynamic DL-driven IDS based on the NSL-KDD dataset, including attribute extraction using the stacked autoencoder (SAE) and classification using soft-max regression (SMR). This model's precision rate (99.27%) demonstrated that the SAE performed better as a DL than traditional superficial models. The authors of [35, 36] both demonstrated that DL algorithms outperformed superficial ML models. Later, [37] used a triangle-area-based method to fasten the procedure of attribute extraction in Multivariate Correlation Analysis as an initial effort at DoS identification (MCA). Using the information that reached the target network, various features were intended to minimize redundancy. To improve the precision of zero-day attack detection, the "triangle region map" module was used to derive the geometrical similarities of two separate functions. The network traffic was converted into images using MCA attribute extraction and analyzed to detect abnormalities. The precision metric of their analyses was 99.95% and 90.12% using sample-wise correlation. For MiTM assaults, a variety of alternative technologies have been suggested. The reviewers increased efficiency by resolving the RNN algorithm's vanishing gradient problem. Initially, the expected value was estimated using a three-month dataset log (for a patient receiving insulin injections). If the expected and estimated values varied by more than a given margin, the right dose could be determined by combining DL and gesture recognition.

5.6 Amalgamation of OpCode and Deep Learning

The component of a machine language command that determines the task to be executed is known as an opcode. For IoT, a blend of OpCode and DL had never been considered before. OpCodes, according to Azmoodeh et al. [27], may be used to distinguish between benevolent and malicious software. 99.68% accuracy was obtained using Eigenspace and deep CNN algorithms, with precision and recall rates of 98.59% and 98.37%, respectively. Wei et al. [30] used the predictive inspection approach to isolate characteristics in order to alleviate malware. The author used system-structured classification to train and validate a classifier.

5.7 *Federated AI Learning*

Generally, the majority of AI models are developed using data from a solitary school of thought in order to safeguard users' identity; this leads to a substandard generalization of the classification algorithm when used with data from other sources. Collaborative AI learning could be the answer to this problem. This method separates the capacity for AI implication in the cloud without mapping it to any confidential health data from the data stored within every hospital by allowing learning from regression/classification analysis that is housed on a public cloud while maintaining the machine learning model in a private network of every institution. Collaborative intelligence was used to categorize distinct levels of the corona pandemic as low, medium, and high risk using blood samples from 1013 individuals from 3 distinct hospitals. Through each facility, 5G was used to transmit user records to a cloud, which then deployed the ML algorithm stored on an outside server to generate the necessary estimations. The cloud-based Automated process only accepts upgrades from each cloud infrastructure, not patient data. This approach correctly classified the seriousness of COVID-19 patient hospitalizations in 95.3%, 79.4%, and 97.7% of the centres, respectively [38].

5.8 *Security Flaw Detection Using Twitter*

Experts from [39] presented a study detailing a new method that examines billions of tweets for references and instances of potential software security susceptibilities and then assesses how much of a concern they pose depending on how they're represented, using ML/DL algorithms. They discovered that Twitter could not just anticipate the multitude of known vulnerabilities that would appear days later on the National Vulnerability Database—the official register of security flaws maintained by the National Institute of Standards and Technology—but also that they could still use NLP to approximately determine if any of those flaws will be assigned a “high” or “critical” rating. They have stated that the version only upgrades once a day, contains some duplicate records, and WIRED's tests skipped some bugs that were later discovered in the NVD. But the research's true breakthrough is its ability to reliably rank the seriousness of vulnerabilities using an automatic interpretation of human language. That implies it could one day act as an efficient information source of new evidence for security professionals seeking to keep their systems safe, or at the very least a part of corporate susceptibility data feeds, or an additional, free feed of threats—weighted for importance—for those administrators to decide. Premised on the NVD's original severity index, the algorithm was able to estimate the severity of the 100 most serious data breaches with 78% precision. They were able to forecast the magnitude of the glitches with 86% accuracy for the first 50, and 100% accuracy for the NVD's ten most serious vulnerabilities.

6 Secure, Private, and Robust Blockchain Techniques as a Countermeasure Against IoT Threats

6.1 Cyber-Physical Systems (CPS)

Machado et al. [40] divided their BC framework into multiple stages: IoT, Fog, and Cloud, with the aim of providing secure communication for CPS. The Trustful Space-Time Protocol (TSTP), which is centered on Proof-of-Trust (PoT), was used to establish trust between IoT devices in a similar realm at the first level. Proof-of-Luck (PoL) was implemented at the Fog stage to construct fault-tolerant IoT data that generates a crypto index for a data audit. The data from the first stage was parsed with a widely implemented hashing scheme known as SHA-256 and stored for the time being. The data was indefinitely deposited at the third level of the cloud, which is a shared database after the acknowledgment and agreement were reached. In addition to data integrity, the analysis provided key maintenance through time modulation and node positioning. TSTP supported clock synchronization, and HECOPS was used to approximate the node's position using inter-iteration. Multiple meta-analyses, such as PoT and PoL, were suggested in the report, but no consideration was given to consumer privacy. One such paper [41] discussed data integrity and the possibility of employing public BC to secure data obtained from drones.

6.2 Prevention of MiTM and DoS Attacks

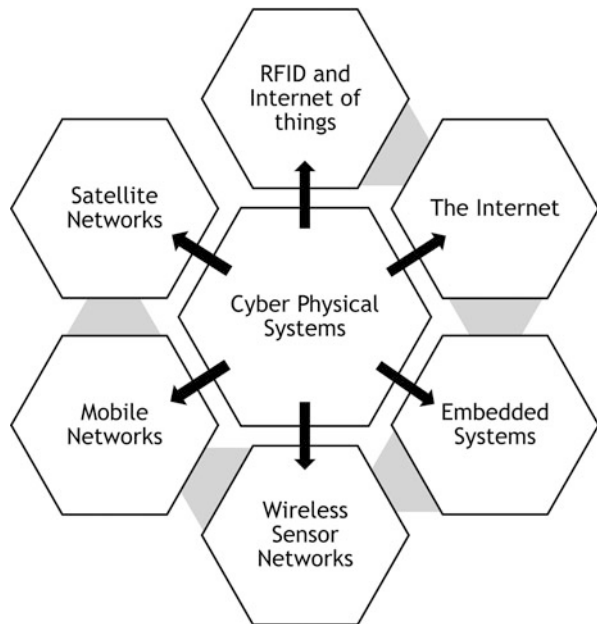
Owing to their relatively simple functionality and the ever-increasing number of vulnerable electronic devices, DoS attacks are one of the most extensively encountered threats. Cybercriminals can quickly monitor several IoT devices to initiate an attack thanks to low-cost IoT technology. The SDN top layer is susceptible to brute force threats, according to [42]. Since SDN is managed by machines, it can be attacked by fraudulent programs and is vulnerable to DoS and DDoS attacks. DDoS prevention approaches used in the past are incompatible with a lightweight multi-standard IoT framework. Given the absence of authorization in the plain-text TCP channel, SDN is vulnerable to floods, saturation, and MiTM threats. Since BC was distributed, lenient, and tamper-proof, Tselios et al. [42] claimed that it was a safer option for protecting IoT devices from potential threats and enforcing confidence amongst multi-vendor devices. These essential BC characteristics render it impervious to data manipulation and flooding assaults. Many of the methods listed above, however, were scientific concepts with no realistic application. Sharma et al. [43] strengthened the design flaw in SDN by introducing DistBlockNet, a decentralized SDN interface for IoT that uses BC. The BC was used to confirm, import, and check the most recent dynamic route table for IoT forwarding applications. The results were very promising; however, in a

separate report, the researchers discovered a MiTM security flaw, thus allowing any cyber-criminal to alter user data sent via Internet [44]. Second, consumers were unable to review their expensive power bills since the new smart grid was unreliable and did not include any early alerts to customers indicating increased electricity consumption. This study suggested implementing encrypted data transfer using shared keys for the username and password as well as the smart agreement which was put on a BC, to prevent the above problems. This method guaranteed a smart-grid infrastructure that was permanent, stable, and straightforward. PoW, on the other hand, maybe highly costly and resource-intensive.

6.3 Intrusion Detection Systems

IDS is a popular surveillance tool for detecting unusual traffic patterns. According to Golomb et al. [45], new anomaly IDS are ineffective since the training process only considers innocuous traffic. An attacker may take advantage of this flaw by inserting malignant data into the system, which would appear to be harmless. Second, the qualified model may be inefficient since it may overlook any IoT interface traffic that is only triggered by an incident, such as a fire alarm. The odds of a hostile attack were slim since a vast range of IoT devices were being educated based on their local data flow. The independent block created for each IoT model, on the other hand, would add to the volume of data (Fig. 8).

Fig. 8 Cyber-Physical Systems (CPS)



6.4 *Privacy Efforts*

In a crowd-sensing program, a BC-based paradigm to address MiTM intervention concerns was also put forth in which user privacy was achieved by the use of the node cooperation process. The authors suggested k-anonymity to ensure user-data privacy, in which the sensing mission was assigned to a community rather than a person. Later, Lu et al. suggested another VANET implementation in [46], in which the developers used the lexicographic Merkle tree to bring anonymity to clients on the current bitcoin network. In addition, falsification was prevented by assigning a prestige weight to each vehicle in the network. Using multi-signatures, the authors of [47] discussed the concerns of payment security and privacy. Since conventional networks were vulnerable, inefficient, and open to the public, messages were transmitted in a cryptographic format which provided protection and protection in correspondence. The public key and private key are used to guarantee user privacy. Similarly, Guo et al. [46] discussed another definition of multi-signatures. The new Electronic Health Record (EHR) system was discovered to be centralized, with no consumer protection or power. Health reports are important papers that provide information about a person's medical condition. They should be regulated by the consumer, but they should also be unalterable. In prior reports, Attribute-Based Signatures (ABS) allowed confidence amongst the parties involved however, it was inaccurate and confined to a singular signature. Owing to the ABS benefits, Guo et al. proposed an ABS with multiple access (MA-ABS), which promised anonymity with controls for accessibility to the recipient, and confidentiality of actual details to the authenticator [46]. Furthermore, using BC for data management boosted immutability and dispersion. MA-ABS was used to maintain confidentiality and a pseudo-random feature seed was used to prevent conspiracy threats. The research also recommended using KeyGen to handle keys.

6.5 *Countermeasures Against Other Attacks*

Several reports have focused on offering alternatives to numerous attacks, in addition to findings on commonly studied security vulnerabilities such as data integrity, MiTM, and DoS. In [48], Sharma et al. proposed a cost-effective, stable, and always-available BC methodology for decentralized cloud infrastructure. The fog nodes' protection was introduced using a mix of SDN and BC. The research moved resource-intensive activities nearer to the boundaries of an IoT network, which increased both security and end-to-end communication latency. The researchers believed that the model was resilient to risks and cyberattacks and the key contribution of this study was to develop a versatile, stable, robust, and fast fog computing infrastructure focused on BC-cloud. In terms, of performance responsiveness, and false alarm rate the distinction was made. Data privacy, management of login credentials, and key control, on the other hand, were not taken

into account. Similarly, Sharma et al. reported in [43] that due to its clustered nature, the current Distributed Mobile Management (DMM) lacked resiliency against cyberattacks. Their suggested BC-based strategy reduced latency and power usage while maintaining the current network architecture. The research, however, relied on PoW consensus, which is energy-driven and has little consumer protection.

7 Future Limitations Concerning Network Security and IoT Devices

1. The gait data gathered by digital sensors embedded in the medical wearables is inextricably connected to the identification of the user, and its accessibility poses a risk to their safety and confidentiality [31]. The techniques for generic user privacy might guarantee that the data is delivered in a format other than plaintext. Nonetheless, there are still certain impediments encountered while implementing this approach. These shortcomings can be used by the perpetrators to decrypt ciphertexts. For instance, the access logs and information on the channel may be protected from manipulation using distributed ledgers. Yet, network insiders have the power to alter or remove crucial information from data, which might cause serious security and integrity issues [31].
2. Insecure Network – A variety of gadgets and software services depend primarily on wireless communication such as Wi-Fi, because of their simplicity and affordability. However, wireless networks are considered to be vulnerable to multiple contraventions, including unauthorized router access, man-in-the-middle attacks, phishing attacks, denial of service, brute-force attacks, jamming, wormhole attacks, crypto-attacks, and traffic injections. Furthermore, the majority of unaccredited free Wi-Fi networks in public areas are untrustworthy which can open the network to unwarranted cybersecurity susceptibilities.
3. Resource Management – IoT, cloud storage, health networks, and all related services are merged into a single stratified framework. When an IoT-cloud-based e-Health infrastructure is deployed, resource management is crucial to consolidate services, remove supererogatory resources, improve efficiency, and reduce system latency. To achieve sustained efficiency improvements and/or avoid loss of efficiency, the resource management system must be continuously configured.
4. Lightweight conventions for devices – To deliver utilities, low-cost systems and software apps consisting of sensors should adhere to strict policies and proxy laws. At the moment, we must use high-cost technologies if we want to provide high-grade protection for sensors. The MIoT scheme, however, poses a disagreement. This problem is encountered due to the fact that the key task of system security is to develop different layers of safety protocols, especially lightweight security protocols, depending on application specifications.

5. **Data Analysis** – A fully automated IoT-cloud-based e-Health infrastructure has tons of integrated mobile sensors that are constantly storing and transmitting a massive amount of data. To interpret all of the data obtained and gathered by IoT-cloud-based e-Health systems, the data processing architecture must expand commensurately to the amount of data collected.
6. **Process of transition** – e-Health replaces or adds controls, medical equipment, and operational protocols to current health networks using IoT-cloud-based systems. New appliances and techniques are seldom easy to integrate. It must be achieved slowly and carefully, and all necessary staff must be well-trained. All new devices and protocols must still be backward compatible with the system that is being replaced or modified.
7. **Security and Privacy** – Systems that use IoT-cloud-based e-Health services must be well aware of the potential susceptibilities and threats to their networks and calculatingly strategize security and privacy structural design to protect them. All susceptibilities and risk vectors must be considered for each device layer, and security and privacy problems must be identified and resolved proactively. Some doctors and medical professionals tend to maintain medical information on servers or local networks that are not wired to the Internet due to confidentiality and privacy issues. Medical record sharing demands the development of technologies that enable physicians to share clinical data while preserving security.
8. **Device performance** – IoT-cloud-based e-Health services used in hospitals, clinics, and other locations can be effective and long-lasting. System performance constraints can prevent technological advancements from being implemented, necessitating an update to improve functionality. This may be the result of a regulatory framework or a system's slow implementation of a specification. The need for knowledge sharing is reduced due to the overlap of roles among the needs of many regions.
9. **Compatibility**: Potential adopters perceive technologies that are compatible with their beliefs, prior interactions, and preferences. Healthcare professionals' aspirations for e-Health services based on IoT and cloud computing should be consistent with the demands of their employment, which include the need to encourage patients and other users to embrace and understand how to use such technology.

8 Future Limitations Concerning ML/BC Approaches

1. **Computation, processing, and data storage**: It is a well-established fact that ML algorithms work best with larger datasets. The addition of data to BC platforms, on the other hand, would deteriorate their output [49]. Finding an equilibrium, which would be optimal for IoT implementations, is an open research topic.
2. **Latency issues**: An IoT network can produce a large amount of data, necessitating more time for training and processing, ultimately increasing the overall output (i.e., latency) of standard machine learning models [50].

3. **Scalability:** In regards to both manufacturing and connectivity costs, ML and BC face scalability issues. With the anticipated growth in data for most IoT networks, many ML algorithms place additional computing and connectivity costs. Similarly, as the percentage of users and networking nodes grows, the BC performs poorly. In typical IoT implementations, where thousands of transactions occur per second, an Ethereum BC averages 12 transactions per second, which is inappropriate [51].
4. **Interoperability and Uniformity:** One of BC's problems, like any new technology, is standardization, which necessitates changes in the law [52]. Cybersecurity is a difficult problem to solve, and it will be blind to reality to assume that we would quickly see a security and privacy norm that would remove all chances of cyber-attack on IoT devices. Nonetheless, a security framework will ensure that computers follow "acceptable" security and privacy requirements. Any IoT computer should have a range of basic protection and privacy capabilities.
5. **Vulnerability:** Although combining ML and BC will greatly improve security and privacy, it does come with some drawbacks. The growing number of risks, such as ransomware and malicious code, makes tracking, detecting, and stopping them in real-time IoT networks more difficult. The training process of machine learning takes longer, and while harmful activity can be detected, it can only be done with a trained model. On the other hand, BC can ensure data non – repudiation and define data transitions.
6. **Risks posed by the momentum of AI development:** AI is taking up a larger share of the conversation each day in terms of raising privacy and security issues, particularly in an age where cyberattacks are widespread and a patient's PHI is highly valuable to identity thieves and cybercriminals. Because of its dualistic structure, AI poses significant challenges: it necessitates and relies on having access to vast volumes of information whilst being vulnerable to data breaches. Because of its rapid progression, existing privacy and security policies and guidelines do not account for AI capabilities. For instance, when ML algorithms can re-identify a log from as little as three data points, current approaches for de-identification are counterproductive at best. AI relies on having access to vast volumes of info. Although, ensuring access to this data, is an impediment both technically and legally. Medical evidence cannot be compartmentalized; it must be paired with other data sources, such as those that provide information on a patient's living circumstances. Simultaneously, the sheer scale, volatility, and fragile nature of the personal data being gathered entails the development of newer, more expansive, reliable, and long-lasting computing infrastructure and algorithms.

9 Future Scope

It is remarkably astonishing that 5G communication was not directly implicated in the creation of nano-embedded systems for use in healthcare, with a focus

on medication delivery for Nano oncology and how conceivably enormous data sets may be analyzed instantaneously by AI to alter the dose. Through the use of 5G, these NEMS may link to other remote systems (wearable tech, cellular telephones, other Microelectromechanical systems (mems, etc.) and develop novel IoNT) services.

10 Conclusion

Although the prospects for the deployment of IoT, ML, DL, and BC applications in the realm of healthcare are limitless, there are many roadblocks in the way, including technology complexities, privacy and security concerns, and a lack of professional expertise. Studies in this domain consider cyber threats and data vulnerabilities to be major impediments. In this chapter, we have succinctly encompassed some promising work from around the globe. In the sense of healthcare confidentiality, we've already discussed various challenges at each stage of the big data lifecycle, as well as the benefits and drawbacks of emerging technology.

We have combed through the latest research to determine the prevailing traits and benefits of IoT-cloud-based e-Health applications, as well as the factors that drive their adoption. We have looked at how various software are used in health systems and how it has brought about a significant change, especially with the implementation of IoT devices and cloud computing. We have also offered an outline of privacy and security issues in these systems generated due to data breaches, malware, and cyberattacks. The prevailing research on e-health systems using ML, DL, and BC techniques has also been presented in detail, by emphasizing their discrepancies explicitly. The IoT network relies heavily on data creation, retrieval, evaluation, and collaboration. An integrative strategy is required with initiatives such as compliance with best practises and continuous testing is used to create a vulnerability-free system. Since fraudulent practices are unpredictable, the device should be able to learn and respond to new threats (zero-day attacks). In this respect, ML/DL can be highly useful in evaluating traffic. Simultaneously, in an IoT setting, the BC can be used to keep track of logging and correspondence. Because this data is unalterable, it can be used as testimony in a court of law with confidence.

In addition, with IoT's exponential growth, the more volume, the poorer the performance. As a consequence, privacy-preserving methods need not have a greater impact on data consistency in order for academics to get the best results. To take it a step forward, we'll attempt to tackle the challenge of balancing protection and privacy frameworks by modeling a variety of methods to help with decision-making and strategic planning.

References

1. Latif, S., Asim, M., Usman, M., Qadir, J., & Rana, R. (2018). *Automating motion correction in multishot MRI using generative adversarial networks*.
2. Aksu, H., Uluagac, A. S., & Bentley, E. (2018). Identification of wearable devices with bluetooth. In *Transactions on sustainable computing (2018)* (p. 1).
3. Zhou, Y., Han, M., Liu, L., He, J. S., & Wang, Y. (2018). *INFOCOM 2018 conference on computer communications workshops*.
4. Kshetri, N. (2017). Blockchain's roles in strengthening cybersecurity and protecting privacy. *Telecommunications Policy*, *41*, 1027–1038.
5. Giles, M. (2019). *Five emerging cyber-threats to worry about in 2019*.
6. Marinov, B., Georgiou, E., Berchiolli, R. N., Satava, R. M., Cuschieri, A., Moglia, A., & Georgiou, K. (2022). 5G in healthcare: From Covid-19 to future challenges. *IEEE Journal of Biomedical and Health Informatics*, 4187–4196. IEEE.
7. de Aguiar, A. W. O., Fonseca, R., Muhammad, K., Magaia, N., Ribeiro, I. D. L., & de Albuquerque, V. H. C. (2021). An artificial intelligence application for drone-assisted 5G remote e-health. *IEEE Internet of Things Magazine*, *4*, 30–35. IEEE.
8. Pasha, M., & Shah, S. M. W. (2018). Framework for e-health systems in IoT-based environments. *Wireless Communications and Mobile Computing*, *2018*, 1–12.
9. Robinson, Y. H., Presskila, X. A., & Lawrence, T. S. (2017). Utilization of internet of things in health care information system. *Internet of Things and Big Data Applications*, *180*, 35–46.
10. Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M., & Liljeberg, P. (2018). Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach. *Future Generation Computer Systems*, *2018*, 1–5.
11. Islam, M. S., Humaira, F., & Nur, F. N. (2020). Healthcare applications in IoT. *Global Journal of Medical Research: (B) Pharma, Drug Discovery, Toxicology & Medicine*, *2020*, 1–3.
12. Shewale, A. D., & Sankpal, S. V. (2020). *IoT raspberry Pi based smart and secure health care system using BSN* (pp. 506–510).
13. Aliverti, A. (2017). Wearable technology: Role in respiratory health and disease. *Breathe*, *13*(2), e27–e36.
14. Collins, A., & Yao, Y. (2018). Machine learning approaches: Data integration for disease prediction and prognosis. *Applied Computational Genomics*, *2018*, 137–141. Springer.
15. Afshar, P., Mohammadi, A., & Plataniotis, K. N. (2018). Brain tumor type classification via capsule networks. In *25th IEEE international conference on image processing (ICIP)* (pp. 3129–3133). IEEE.
16. Zhu, W., Liu, C., Fan, W., & Xie, X. (2018). Deeplung: Deep 3D dual path nets for automated pulmonary nodule detection and classification. In *2018 IEEE winter conference on applications of computer vision (WACV)* (pp. 673–681). IEEE.
17. Ibrahim, M., Chakrabarty, K., Firouzi, F., & Farahani, B. (2018). From EDA to IoT e-health: Promises, challenges, and solutions. In *IEEE transactions on computer-aided design of integrated circuits and systems*. IEEE.
18. Cao, N., Wang, C., Li, M., Ren, K., & Lou, W. (2011). Privacy-preserving multi-keyword ranked search over encrypted cloud data. In *Proceedings of the IEEE INFOCOM* (pp. 829–837).
19. Bezawada, B., Liu, A. X., Jayaraman, B., Wang, A. L., & Li, R. (2015). Privacy preserving string matching for cloud computing. In *Proceedings of the 35th IEEE international conference on distributed computing systems, ICDCS '15* (pp. 609–618). IEEE.
20. Jing, Q., Vasilakos, A., Wan, J., Lu, J., & Qiu, D. (2014). Security of the internet of things: Perspectives and challenges. In *Wireless networks 20 (11 2014)* (pp. 2481–2501).
21. Makhdoom, I., Abolhasan, M., Lipman, J., Liu, R. P., & Ni, W. (2019). Anatomy of threats to the internet of things. In *IEEE communications surveys tutorials 21, 2 (Secondquarter 2019)* (pp. 1636–1675). IEEE.

22. Brass, I., Tanczer, L., Carr, M., Elsdon, M., & Blackstock, J. (2018). Standardising a moving target: The development and evolution of IoT security standards. *Living in the Internet of Things: Cybersecurity of the IoT, 2018*, 1–9.
23. Milosevic, J., Malek, M., & Ferrante, A. (2016). A friend or a foe? Detecting malware using memory and CPU features. In *Proceedings of the 13th international joint conference on e-business and telecommunications (ICETE 2016)* (Vol. 4, pp. 73–84).
24. Jing, X., Yan, Z., Jiang, X., & Pedrycz, W. (2019). Network traffic fusion and analysis against DDoS flooding attacks with a novel reversible sketch. *Information Fusion, 51*(2019), 100–113.
25. Elejla, O. E., Belaton, B., Anbar, M., Alabsi, B., & Al-Ani, A. K. (2019). Comparison of classification algorithms on icmpv6-based DDoS attacks detection. *Lecture Notes in Electrical Engineering, 481*(2019), 347–357.
26. Rezaad, M., Brust, M. R., Akbari, M., Bouvry, P., & Cheung, N. M. (2018). Detecting target-area link-flooding DDoS attacks using traffic analysis and supervised learning. *Advances in Information and Communication Networks, 2018*.
27. Azmoodeh, A., Dehghantanha, A., & Choo, K.-K. R. (2018). Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning. *IEEE Transactions on Sustainable Computing*.
28. Aonzo, S., Merlo, A., Migliardi, M., Oneto, L., & Palmieri, F. (2017). Low-resource footprint, data-driven malware detection on android. *IEEE Transactions on Sustainable Computing, 3782*.
29. Feng, P., Ma, J., Sun, C., Xu, X., & Ma, Y. (2018). A novel dynamic android malware detection system with ensemble learning. *IEEE Access, 6*(2018), 30996–31011.
30. Wei, L., Luo, W., Weng, J., Zhong, Y., Zhang, X., & Zheng, Y. (2017). Machine learning-based malicious application detection of android. *IEEE Access, 5*(2017), 25591–25601.
31. Liu, J., Bi, H., & Kato, N. (2022). Deep learning-based privacy preservation and data analytics for IoT enabled healthcare. *IEEE Transactions on Industrial Informatics, 18*, 4798–4807. IEEE.
32. Restuccia, F., DrOro, S., & Melodia, T. (2018). Securing the internet of things in the age of machine learning and software-defined networking. *IEEE Internet of Things Journal, 1*, 1–14.
33. Chaabouni, N., Mosbah, M., Zemmari, A., Sauvignac, C., & Faruki, P. (2019). Network intrusion detection for IoT security based on learning techniques. *IEEE Communications Surveys Tutorials 21, 3 (Thirdquarter 2019)*, 2671–2701.
34. Sharmeen, S., Huda, S., Abawajy, J. H., Ismail, W. N., & Hassan, M. M. (2018). Malware threats and detection for industrial mobile-IoT networks. *IEEE Access, 6*(2018), 15941–15957.
35. Diro, A., & Chilamkurti, N. (2018). Leveraging LSTM networks for attack detection in fog-to-things communications. *IEEE Communications Magazine, 56*, 124–130.
36. Abeshu, A., & Chilamkurti, N. (2018). Deep learning: The frontier for distributed attack detection in fog-to-things computing. In *IEEE Communications Magazine, 56*, 169–175.
37. Tan, Z., Jamdagni, A., He, X., Nanda, P., & Liu, R. P. (2014). A system for denial-of-service attack detection based on multivariate correlation analysis. *IEEE Transactions on Parallel and Distributed Systems, 25*, 447–456.
38. Ma, Y., Talha, M., Al-Rakhami, M. S., Wang, R., Xu, J., & Ghoneim, A. (2021). Auxiliary diagnosis of Covid-19 based on 5G-enabled federated learning. *IEEE Network, 35*, 14–20. IEEE.
39. Zong, S., Ritter, A., Mueller, G., & Wright, E. (2019). *Analyzing the perceived severity of cybersecurity threats reported on social media*.
40. Machado, C., & Frohlich, A. A. (2018). IoT data integrity verification for cyber-physical systems using blockchain. In *Proceedings – 2018 IEEE 21st international symposium on real-time computing, ISORC 2018* (pp. 83–90).
41. Liang, X., Zhao, J., Shetty, S., & Li, D. (2017). Towards data assurance and resilience in IoT using blockchain. In *MILCOM 2017 – IEEE military communications conference (MILCOM)* (pp. 261–266).
42. Tselios, C., Politis, I., & Kotsopoulos, S. (2017). Enhancing SDN security for IoT-related deployments through blockchain. In *IEEE conference on network function virtualization and software defined networks, NFV-SDN 2017, January* (pp. 303–308).

43. Sharma, P. K., Singh, S., Jeong, Y. S., & Park, J. H. (2017). Distblocknet: A distributed blockchains-based secure SDN architecture for IoT networks. *IEEE Communications Magazine*, 55(9), 78–85.
44. Gao, J., Asamoah, K. O., Sifah, E. B., Smahi, A., Xia, Q., Xia, H., Zhang, X., & Dong, G. (2018). Gridmonitoring: Secured sovereign blockchain based monitoring on smart grid. *IEEE Access*, 6(2018), 9917–9925.
45. Golomb, T., Mirsky, Y., & Elovici, Y. (2018). *Ciota: Collaborative IoT anomaly detection via blockchain*.
46. Guo, R., Shi, H., Zhao, Q., & Dong, Z. (2018). Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems. *IEEE Access*, 6(2018), 11676–11686.
47. Aitzhan, N. Z., & Svetinovic, D. (2018). Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. *IEEE Transactions on Dependable and Secure Computing* 15, 5, 4, 840–852.
48. Sharma, P. K., Chen, M. Y., & Park, J. H. (2018). A software defined fog node based distributed blockchain cloud architecture for IoT. *IEEE Access*, 6, 115–124.
49. Song, J. C., Demir, M. A., Prevost, J. J., & Rad, P. (2018). Blockchain design for trusted decentralized IoT networks. In *2018 13th system of systems engineering conference*.
50. Dorri, A., Kanhere, S. S., & Jurdak, R. (2016). *Blockchain in internet of things: Challenges and solutions*.
51. Khaled Salah, M., Rehman, H. U., Nizamuddin, N., & Al-Fuqaha, A. (2019). Blockchain for AI: Review and open research challenges. *IEEE Access*, 7, 10127–10149.
52. Niwa, H. (2007). *Why blockchain is the future of IoT?*

GPU Based AI for Modern E-Commerce Applications: Performance Evaluation, Analysis and Future Directions



Sanskar Tewatia, Ankit Anil Patel, Ahmed M. Abdelmoniem, Minxian Xu, Kamalpreet Kaur, Mohit Kumar, Deepraj Chowdhury, Adarsh Kumar, Manmeet Singh, and Sukhpal Singh Gill 

Abstract Every year in the United States, the 4th Thursday of November is commemorated as Thanksgiving Day. The next day which is a Friday is known as Black Friday. This day is the busiest day in terms of shopping because all major retailers and e-commerce websites offer massive amounts of discounts and deals. Hence, this sale is termed the Black Friday Sale. There is a lot of potentials to make a profit even after such discounts if the sales patterns from previous years' data are analyzed properly. Investigating various demographics of customers and analyzing the purchase amount spent by each customer on various products, there is a need

S. Tewatia

Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, USA

A. A. Patel · A. M. Abdelmoniem · S. S. Gill (✉)

School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

e-mail: s.s.gill@qmul.ac.uk

M. Xu

Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

K. Kaur

Seneca International Academy, Toronto, Canada

M. Kumar

Department of Information Technology, National Institute of Technology, Jalandhar, India

D. Chowdhury

Department of Electronics & Communication Engineering, International Institute of Information Technology (IIIT), Naya Raipur, India

A. Kumar

Department of Systemics, School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India

M. Singh

Centre for Climate Change Research, Indian Institute of Tropical Meteorology (IITM), Pune, India

Jackson School of Geosciences, University of Texas at Austin, Austin, TX, USA

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

M. Kumar et al. (eds.), *6G Enabled Fog Computing in IoT*,

https://doi.org/10.1007/978-3-031-30101-8_3

to find out patterns for this behavior. Therefore, we utilized classical and modern artificial intelligence and machine learning techniques such as Linear Regression, Neural Networks, Gradient Boosting Trees and AutoML, to make predictions on the available test data to find a model for the most accurate predictions. We used a Graphical Processing Unit (GPU)-based high-performance computing environment to analyze the performance of various artificial intelligence and machine learning techniques for e-commerce applications. Since the dataset contains information about various demographics and backgrounds of the customers, we encoded the data in an easy-to-understand format and reduce the bias for each algorithm. Furthermore, various techniques of feature engineering are used, and new features are generated from existing features by grouping the target variable purchase, for each sub-category of that feature. Erroneous predictions are also handled; ultimately, the model performed well on unseen test data. Finally, this study can help researchers to find the best model with more accurate predictions for e-commerce applications.

Keywords Artificial intelligence · Machine learning · Black Friday sales · High-performance computing · Graphical processing unit

1 Introduction

Earlier, the sales used to start on Thanksgiving Day (Thursday), but over time, more and more retailers began starting their sales later of the day. Nowadays, the sale has been termed as Black Friday Sales because all sellers start the sales on Friday only [1, 2]. There is a lot of scope for increasing the profits of retailers during such sales [3]. This can be done by analysing the type of products that customers are buying, what kind of products to target to a specific demographic of customers, and recommending products based on the average expenditure of customers from various backgrounds, for various categories of products [4]. Trends from such analysis can be used to create recommender systems, in order to market the products better so that the particular product comes up at the top while searching for them on online websites. Such analysis also helps characterize the quantity of particular product that needs to be stocked so as to meet the demand [5].

In our paper, we meticulously analyse a dataset shared by Analytics Vidhya for an ongoing global contest [6], where one has to apply machine learning models and predict the number of purchases done by customers during the upcoming Black Friday sales, by analysing the trends of purchase amount depending upon the available demographics of customers and the product categories. This data is obtained from the purchase summary dataset for a few categories of high-volume products from a previous sale. We also present various data pre-processing techniques that are not specific to this dataset but can also be utilized for other datasets. These techniques immensely help in improving the accuracy of machine learning models [7]. We also utilized one more technique which is feature augmentation. Feature

augmentation is a technique to feature augmentation helps to enhance the breadth of seen domains for training [8]. Since only 11 features were available to us, new features were generated from these 11 features to extract valuable information about each customer, product, and product category [6]. Such feature generation techniques can be employed in other machine learning approaches where very limited data is available [9].

Hence, after careful analysis of customer and product data, machine learning models were used to predict the purchase amount for old or new customers and products. It was observed that the best results were obtained from the extensions of Gradient Boosting Trees – CatBoost and XGBoost [10, 11]. To further improve the score, weighted ensembles can be taken from our own previously generated predictions [12]. The main outcome from the prediction of such a problem is that the retailers can now stock the products appropriately and provide deals accordingly to maximise the profit earned [13]. Hence, this solution is not limited to Black Friday sales but can also be used for purchase prediction or demand analysis problems as well as other market related problems [14].

1.1 Motivation and Our Contributions

Retail sales are in fact of the most crucial field for research and exploitation for the domain of data analytics in business. Black Friday sale is one of the most impactful sales on a global level. There are multiple preconceived notions in the minds of the general public, such as women being more interested when it comes to shopping [3]. There can be many such notions [4], but the following analytical questions arise: how do such claims compare to the sales data gathered from retailers or e-commerce websites? Which city from a particular state spends the most on sales? What is the age group of people that spends the most? What effect does marriage have on shopping? How do these factors help in the prediction of sales amount for an upcoming sale? What can the retailers do to maximize their profits? How can they make predictions on sales to stock their shops accordingly? Data analysis on the available data and machine learning techniques can be utilized to find the answers to such questions [6].

In this world of online shopping, e-commerce websites have a lot of information about their customers the offline retail stores might not have [5]. Looking at these demographics and the history of sales of products by customers, it is possible to analyse this data and apply machine learning techniques to predict future sales, so that the retailer can stock their products and price them appropriately. In this paper, data from Black Friday sales of 2016 is thoroughly analysed and numerous feature engineering techniques are used to give this data to machine learning models, which can make predictions on future sales. These techniques can be used for a wide variety of problems by the private sector. And, we shed light on the importance of the model choice for the robustness and speed of modern-day algorithms used by researchers.

This problem is listed in the form of a global contest by Analytics Vidya, where anyone can participate and check their standings with respect to people from all over the world. The dataset is also provided by Analytics Vidhya for the purpose of this contest [6].

When it comes to sales prediction, many factors such as age, gender, marital status, occupation, type of product, city, etc. can immensely impact the purchase volume [13]. In order to make personalized predictions for each customer, these demographics were given in the form of categorical data in this dataset [14]. Using data pre-processing and machine learning techniques, the contest required the participants to submit their results on the contest website.

The evaluation metric for this paper and the public ranking used was Root Mean Squared Error (RMSE). RMSE is an extremely popular metric when training machine learning models and deep neural networks [12]. RMSE is essentially the standard deviation of the prediction errors. Prediction errors are the difference between actual/theoretical values and predicted/experimental values. RMSE is obtained by squaring the prediction errors, taking their mean, and finally taking their squared root as defined below [15]:

$$RMSE_{fo} = \left[\sum_{i=1}^N (Z_{fi} - Z_{oi})^2 / N \right]^{1/2} \quad (1)$$

- \sum – Summation for all data entries
- $(Z_{fi} - Z_{oi})^2$ – Square of Prediction Error
- N – Sample Size

For this contest, 30% of the test data has been posted to the general public in order to get the current global ranking. At the end of the contest, rankings will be based on the RMSE score obtained on the remaining 70% of the test data, which is at present, not public. As of the date of writing this paper, 23,159 people have taken part in this contest by uploading their submissions evaluated on the public 30% test data. To conduct our experiments, we leverage a Graphical Processing Unit (GPU)-based high-performance computing environment to analyze the performance of various artificial intelligence and machine learning techniques for e-commerce applications. The proposed techniques led to our submission securing a decent rank of 119 out of a total of 23,159 participants based on the Root Mean Squared Error (RMSE) on the Black Friday Sale public test data of the contest.

The major contributions of this research work are:

1. The analysis of Thanksgiving day sales and product demand, alongside purchase prediction will help retailers make predictions about the demand for these products.
2. Leveraging a Graphical Processing Unit (GPU)-based high-performance computing environment is used to analyse the performance of various artificial intelligence and machine learning techniques for e-commerce applications.

3. Developing a machine learning model for making predictions about the upcoming Black Friday sale, and comparing multiple models with each other based on their Root Mean Square Error (RMSE) and the actual values.
4. We show that our techniques led to significant improvements in models' performance which secured our submission a rank of 119 out of 23,159.

1.2 Article Organisation

This book chapter organization has been as follows: Section 2 presents the related work. Section 3 aims to present the background of the problem. Section 4 explains the methodology. Section 5 presents the results and analysis. Finally, Section 6 concludes the paper and presents the possible future scope.

2 Related Work

This problem has been available on Analytics Vidhya since July 2016 [6]. Trung et al. [16] have discussed the implementation of bagging and boosting algorithms for this problem. Their major focus had been on how effectively to divide the dataset for training and testing to train the models. Their experimentation concluded that the scheme of data splitting that resulted in the best predictions was when they chose a 70:30 split for the training and testing data, respectively. They also observed that for this problem, boosting algorithms performed better than bagging-based algorithms, because taking the mean of multiple-week learners were resulting in the formation of a weak combined model. Ching et al. applied various regression-based algorithms namely – Linear, Logistic, Polynomial, Stepwise, Ridge, Lasso & Elasticnet Regression; along with MLK Classifiers, Decision Trees, Deep Learning using Keras, and also XGBoost for this problem. They concluded that models like neural networks were too complex for this kind of a problem, which they showed could be easily outperformed by data cleaning and simpler regression models like decision trees, both with and without bagging, as well as XGBoost.

Kalra et al. [17] studied the effects of modifying the ratio of training and testing data so as to find the model with the best score on a common metric – Root Mean Squared Error (RMSE). The results from their experiments helped them identify models which were overfitting and those which were underfitting. Along with the train-test splitting ratio, they also printed the feature importance from various models and observed that for all the models, 3 specific features had the highest feature importance and were hence, the major contributing attributes for purchase prediction.

Xin et al. [18] studied the development of the e-commerce website on different special days like Black Friday, Cyber Monday, etc. and predicted the sales prediction for online promotion on these special days. Authors have proposed Deep Item

Table 1 Comparison with related works

Components of experimentation		[16]	[19, 20]	[17]	Our study (this work)
Pre-processing	Outlier removal				✓
	OneHot encoding				✓
Features	Product category	✓	✓	✓	✓
	Marital status	✓	✓	✓	✓
	City category	✓	✓	✓	✓
	Occupation	✓	✓	✓	✓
	Age	✓	✓	✓	✓
	Purchasing power				✓
	Count variables				✓
	Min, max & Mean Variables				✓
	25, 50 & 75 percentile values				✓
Handling negative predictions					✓

Network for Online promotion. They have designed the Deep Item Network using a novel target user-controlled gated recurrent unit structure for the dynamic features and provided a new attention mechanism using a static user profile. They have compared this novel mechanism with a traditional prediction algorithm. Wu et al. [19] and Ramasubbareddy et al. [20] proposed a prediction model using different machine learning models like regression, neural network, bagging classifiers, etc. and studied the pre-processing and visualisation techniques. In this paper, the authors have used ridge regression, lasso regression, linear regression, elasticnet regression, logistic regression, stepwise regression and polynomial regression. In machine learning, authors have used linear regression, MLK classifier, Keras-based Deep learning model, Decision Tree, Decision Tree with Bagging, and XGBOOST. As the main evaluation metric, the RMSE of the algorithms are compared in this paper.

2.1 Critical Analysis

Table 1 depicts the critical analysis that has been carried out, as compared to other state of art approaches. A new set of features were created and added, pertaining to unique entries of User_ID, Product_ID or Product_Category. Firstly, the outliers in the dataset were identified and removed, by taking the top 99.5 percentile of the Purchase column in the dataset (Outlier removal). A specific set of data encoding techniques were used in the dataset, for the conversion of all the categorical variables into numerical variables (Onehot Encoding, Numerical Encoding), to be able to use them for training machine learning models. Finally, a technique for handling erroneous predictions was utilized for all the models. All these techniques helped improve the performance of the models.

3 Methodology

In this section, the dataset is described, and various techniques are applied for feature engineering and data augmentation. After pre-processing, this data is fed to various Machine Learning Models and then these algorithms are ranked depending on their RMSE.

3.1 Dataset

“ABC Private Limited” is a retail company whose aim is to understand customer purchase patterns (i.e., purchase quantity) of various products belonging to separate categories. They have decided to share this purchase summary dataset from various customers for a few categories of commonly sold high-volume products from the previous month. This summary reveals various customer demographics. The ultimate aim is to apply machine learning models for prediction of upcoming purchases Sale, so that the company can stock the products appropriately and consequently provide adequate discounts for a better profit margin [6]. For the purpose of this global contest, this dataset was provided by Analytics Vidhya. The dataset consists of 11 features (Table 2), a mix of categorical and numerical features and one target variable – Purchase. There are a total of 550,068 rows of data records in the dataset.

Table 2 Dataset details

Variable	Definition
User_ID	User identity no.
Product_ID	Product identity no.
Gender	Gender
Age	Peer-group
Occupation	Job type
City_Category	Belonging to City (A,B,C)
Stay_In_Current_City_Years	No. of years in city to find out if they are native residents or immigrants
Marital_Status	Married or unmarried
Product_Category_1	First category
Product_Category_2	Second category
Product_Category_3	Third category
Purchase	Purchase value to be predicted

Table 3 Number of unique entries across each feature

User_ID	5891
Product_ID	3631
Gender	2
Age	7
Occupation	21
City_Category	3
Stay_In_Current_City_Years	5
Marital_Status	2
Product_Category_1	20
Product_Category_2	17
Product_Category_3	15
Purchase	18,105

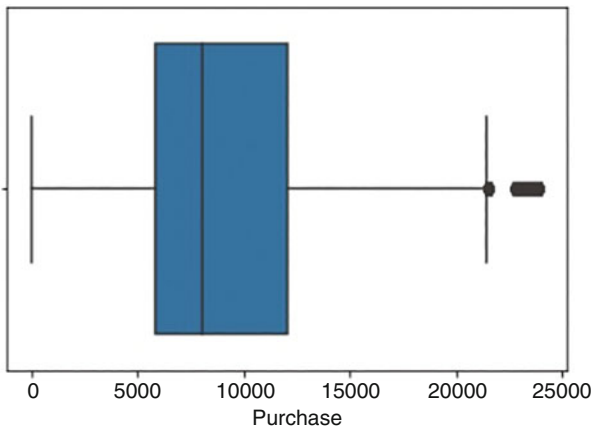


Fig. 1 Boxplot of the target variable – Purchase

3.2 Data Exploration

First, we find out the total number of unique values of each feature. This will tell us how discrete each feature is, because same *User_ID* can buy multiple products and also the fact that the same *Product_ID* can be bought by multiple users.

Table 3 gives us information about the repetitions in the number of *User_IDs* & *Product_IDs*. We can construct data from this existing dataset based on buying power per *User_ID* or mean, maximum, and minimum expenditure for each unique *User_ID*. After looking at the boxplot of the target variable in Fig. 1 (i.e., the purchase column), we observe that most of the data is concentrated around 8000 units, whereas there are very few entries where the value of these Purchases is more than 22,000 units. Hence, these outlier rows need to be taken care of during the data pre-processing stage.

Fig. 2 Average purchase across native and immigrant residents



Fig. 3 Average purchase across each City



Figure 2 shows that the average value of purchase amount across natives and immigrants is not significantly different because the value is around 9000 units for each category for different number of years a customer has been staying in that state.

The data in Fig. 3 from 3 cities depict that again, the average expenditure from customers in each city is between 8500–9500 units and that customers from city C tend to spend the most, while those from city A spent the least purchase (or money) amount.

Figure 4 shows that the average purchase does not differ between married and unmarried customers.

Contrary to popular belief, we find that on average males are actually spending more money on shopping, compared to women (Fig. 5).

Fig. 4 Average purchase by married and unmarried customers



Fig. 5 Average purchase by each gender



Again, data from Fig. 6 demonstrates that irrespective of their age groups, on average, customers tend to spend similar amounts of money during Black Friday Sales.

It is evident from Fig. 7 that there is no significant difference between the value of mean Purchase across various categories of occupations of the customers. Therefore, it can be inferred that on average, people spend very similar amounts of money irrespective of their background, hence the Purchase Column can be considered uniformly distributed.

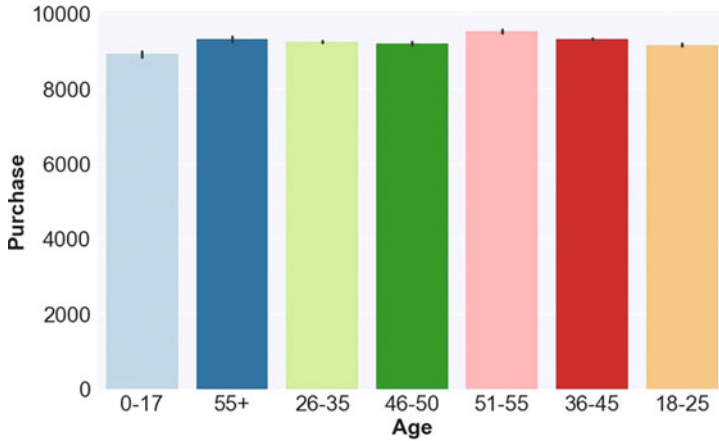


Fig. 6 Average purchase across each age group

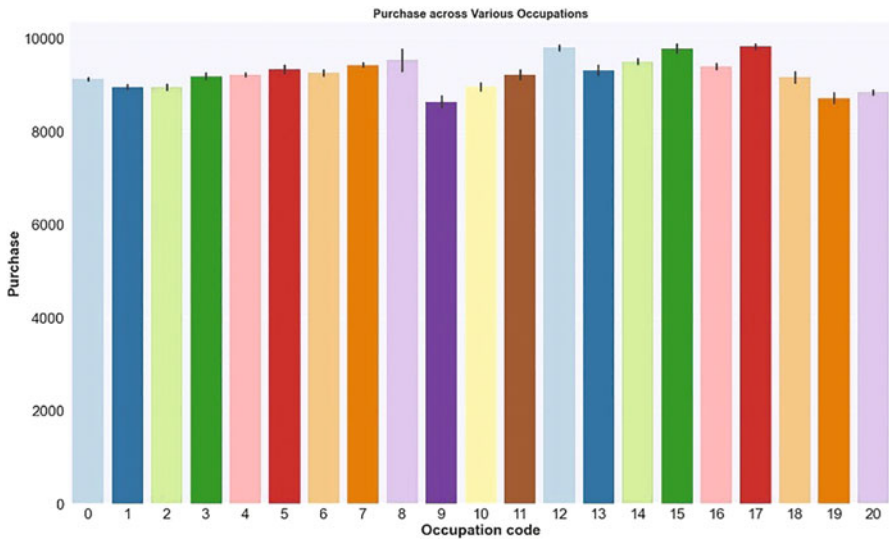


Fig. 7 Average purchase by each occupation

4 Data-Preprocessing

In this section, various techniques of data pre-processing are applied to the dataset so that it is converted into an appropriate format, to be used for model building and training. Figure 8 shows the flow of the data pre-processing.



Fig. 8 Flow of Data Processing

Table 4 Percent of total training data that is missing

User_ID	0
Product_ID	0
Gender	0
Age	0
Occupation	0
City_Category	0
Stay_In_Current_City_Years	0
Marital_Status	0
Product_Category_1	0
Product_Category_2	31.56
Product_Category_3	69.67
Purchase	0

Table 5 Filling null values

Approach	Root Mean Squared Error
Filling with 0	4623.98
Filling with -999	4610.71

4.1 Null Values

Since customers can purchase products from any category they wish, there is a need to fill these rows with some data. If we drop all such rows, we will lose most of our training data and will be unable to build unbiased models.

In order to fill in the missing data (Table 4), two approaches were tried - filling all these rows with Zero and filling them with -999. Two separate models were trained using the same linear regression technique. The evaluation metric used for comparison was Root Mean Squared Error on the test data. It was evident from the RMSE scores in Table 5 that filling all the missing data with -999 showed a slightly better RMSE score compared to the score obtained when filling them with zero, hence we proceed by filling them with a value of -999.

4.2 Removing Outliers

In machine learning, outliers are extreme values that are outside the range of expected observations, and not akin to the remaining data. Hence, it is important to ensure the absence of outliers to improve model fitting and further help towards the prediction of more accurate values. The boxplots of various percentiles of data

Fig. 9 100 Percentiles of data

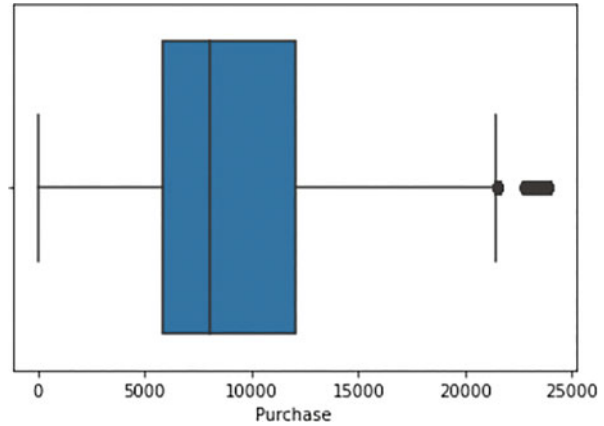
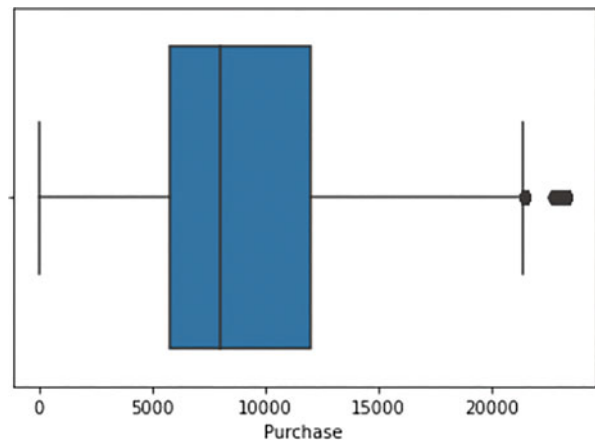


Fig. 10 99.75 Percentiles of data



are shown in Fig. 9 (100 Percentiles of data), Fig. 10 (99.75 Percentiles of data) and Fig. 11 (99.5 Percentiles of data), it was found that taking only the 99.5 percentile of the train data helped get rid of outliers and the new dataset consisting of rows where the value of Purchase remained in this range.

4.3 Feature Generation

Since there were only 11 features for the target variable, new features need to be generated from these existing features to improve model performance. It was evident from Table 2 that User_IDs were highly repetitive. Therefore, we introduce a new feature called “Purchasing_Power” which groups the Purchase column values by User_IDs and then taking sum for every User_ID. In our implementation, a new series was created with this information using Pandas. Taking percentiles

Fig. 11 99.5 Percentile of data

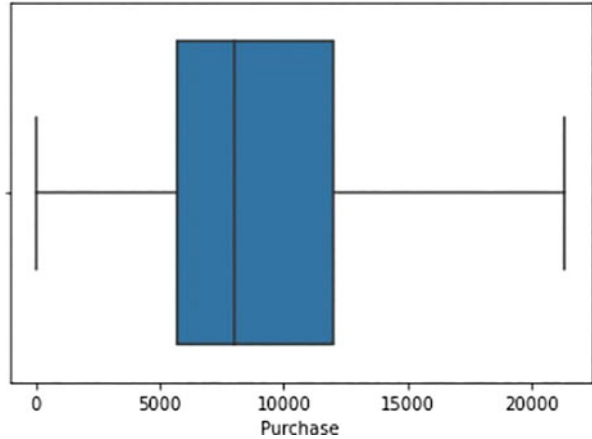


Table 6 New features created from existing features

MinPrice	Minimum value of <i>Purchase</i> column for various categories of that feature
MaxPrice	Maximum value of <i>Purchase</i> column for various categories of that feature
MeanPrice	Mean value of <i>Purchase</i> column for various categories of that feature
25_Percentile	25 Percentile value of <i>Purchase</i> Column for various categories of that feature
50_Percentile	50 Percentile value of <i>Purchase</i> Column for various categories of that feature
75_Percentile	75 Percentile value of <i>Purchase</i> Column for various categories of that feature

(in multiples of 10) of the total sum of this series, a number from 1 to 10 was assigned in a dictionary, to each unique *User_ID*, based on the comparison of each *User_ID*'s expenditure and the percentile of the total sum. Then for each *User_ID* in the train and test set, this dictionary was mapped to a new feature named *Purchasing_Power*. If a new *User_ID* comes up, absent in the dictionary, a mean value of 5 will be assigned as the *Purchasing_Power* to that new *User_ID*. Moreover, count variables were generated for *Product_Category_3*, *Product_ID*, *Age*, *User_ID*, *Product_Category_1* *Occupation*, and *Product_Category_2*. These counters would help identify repeating customers, from a particular age group, occupation, or their unique *User_ID*s or the *Product_ID*s that were bought multiple times. Furthermore, 5 more features were generated for each of the following features – *User_ID*, *Product_ID*, *Product_Category_1*, *Product_Category_2* and *Product_Category_3* as shown in Table 6. This is done by grouping the *Purchase* Column by these 5 features and operating on them. All of these new features were of numerical type. Hence, separate encoding was not required.

Table 7 Type of input for each algorithm

Linear regression	Only numerical input
Neural Networks	Only numerical input
XGBoost	Only numerical input
Catboost	Both numerical & string input
AutoML	Only numerical input
Ensemble Model	Both numerical & string input

4.4 Data Encoding

There are two types of predictors in datasets, when trying to make prediction using machine learning algorithms. The data can either be in numeric form, for e.g., age, height, weight, salary, etc., or categorical form was finite values, where the data row belongs to a particular category, e.g., blood group, nationality, type of degree, marital status, gender, etc. Most machine learning models require the data in the predictors to be of numeric format. Categorical features are handled differently by every algorithm. In total, 6 approaches (Table 7) were tried and in two of them, it was possible to specify the categorical variables to the algorithm beforehand [21]. In others, the categorical variables were converted to numerical format using various kinds of encoding.

The columns titled `User_ID` and `Product_ID` might contain crucial information about customers and specific products hence dropping these columns would have negatively impacted the performance and robustness of the models. Hence, these columns were Label-Encoded to convert them to Integer format. In Label-Encoding, each label is assigned a distinctive integer value based on alphabetical ordering. Hence, for each model, 5891 unique integer values were assigned for each `User_ID`, and 3631 unique integer values for `Product_ID`. The purpose of One-Hot Encoding is to create new features based on the different category values that were present in that specific feature. Thus, a unique value in that category is added to the dataframe object, as a new feature column. For example, in our dataset, the feature `City_Category` is One-Hot Encoded for all approaches. For different approaches, the categorical features were converted to numerical form either using One-Hot Encoding or Label Encoding.

4.5 Handling Erroneous Predictions

After making predictions on the test set, it was observed that some values of `Purchase` were predicted as negative, which is not theoretically possible since the minimum value of `Purchase` in the train dataset was always positive. Hence, in the predicted values of `Purchase`, all negative predictions from all models were replaced with a value of zero, indicating no purchase of that particular product.

4.6 Model Building

Various algorithms were analysed and used for training the dataset and making predictions on the test dataset.

4.6.1 Linear Regression

The motivation behind linear regression is to find out *linear additive* relationships between variables. Let Y denote the “dependent” variable, and its values has to be predicted, and let X_1, \dots, X_k denote the “independent” predictors which are present, from which one wishes to predict Y , with the value of variable X_i in period t (present in the dataset inside row t) denoted by X_{it} . For such a case, the equation for calculating the predicted value of Y_t is as follows in Eq. (2):

$$Y = b_0 + b_1X_{1t} + b_2X_{2t} + \dots .b_kX_{kt} \quad (2)$$

For our dataset, this variable Y is the Purchase feature. Since this model can only take numerical input, the input features – Age, Gender, Stay_In_Current_City were label-encoded. Rest of the features were already numerical.

4.6.2 XGBoost

An extension of the [gradient boosting decision tree algorithm](#) is Extreme Gradient Boosting. This software library is mainly focused on the speed of computation and performance of model. Boosting is an ensemble-based technique in which new models are trained sequentially, and in this process, the errors made by existing/old models are corrected. In this approach, new models are added sequentially, and this process continues until there is no more improvements in the sum of squared residuals [10]. The main advantages of this algorithm are – support for parallel processing, inbuilt cross-validation, and variety of regularization techniques to reduce over-fitting. Gradient boosting assists in the prediction of optimal gradient for additive models, unlike classical gradient descent techniques where output errors are reduced at each iteration. Since this model can only take numerical input [10], the input features – Age, Gender, Stay_In_Current_City were label-encoded. Rest of the features were already numerical.

GridSearchCV is a library function that helps in looping through predefined hyperparameters and fitting of estimators (models) on the training set so as to select the most optimum values of parameters from a predefined list of hyperparameters [22]. Table 8 lists the best value of these hyperparameters for XGBoost, attained after extensive experimentation.

It is clearly evident from Table 9 that the feature importance of the newly generated features, proposed in this work, is much higher compared to that of the original features. Hence, it can be concluded that these features helped the model immensely.

Table 8 Hyperparameter values for XGBoost

Parameter	Value
Eta	0.03
min_child_weight	10
Subsample	0.8
Colsample_bytree	0.7
Max_depth	10
n_estimators	750
Num_rounds	1500

Table 9 Feature importance for XGBoost

Features	Importance (%)	Features	Importance (%)
PID_MeanP	46.78	User_ID	0.52
PID_50Perc	10.87	Age_Count	0.51
PID_75Perc	9.3	Pc3_25Perc	0.51
PID_25Perc	5.16	City_Category_C	0.51
UID_75Perc	1.26	Pc3_75Perc	0.51
UID_MeanP	1.22	Product_Category_2_Count	0.5
UID_25Perc	1.01	Pc2_MinP	0.5
UID_50Perc	0.97	OccupationCount	0.5
Pc1_MaxP	0.9	Pc2_50Perc	0.49
Pc1_50Perc	0.78	Product_Category_3	0.49
PID_MaxP	0.74	Pc2_75Perc	0.49
Pc1_MinP	0.74	Pc3_MeanP	0.49
Gender	0.71	Occupation	0.48
Pc1_25Perc	0.69	UID_Min	0.48
Pc1_75Perc	0.69	Product_Category_1_Count	0.47
Pc1_MeanP	0.66	PID_MinP	0.47
UID_MaxP	0.65	Pc2_25Perc	0.47
User_ID_Count	0.63	Stay_In_Current_City_Years	0.46
Purchasing_power	0.56	Pc2_MeanP	0.46
Age	0.54	Product_ID_Count	0.46
Product_Category_3_Count	0.54	Pc2_MaxP	0.46
Pc3_50Perc	0.54	Product_Category_2	0.44
Pc3_MinP	0.53	Product_ID	0.44
City_Category_A	0.53	City_Category_B	0.43
Pc3_MaxP	0.52	Marital_Status	0.41
Product_Category_1	0.52		

Model: "sequential_1"

Layer (type)	Output Shape	Param #
dense_7 (Dense)	(None, 256)	11520
dense_8 (Dense)	(None, 1024)	263168
dense_9 (Dense)	(None, 1024)	1049600
dense_10 (Dense)	(None, 512)	524800
dense_11 (Dense)	(None, 512)	262656
dense_12 (Dense)	(None, 32)	16416
dense_13 (Dense)	(None, 1)	33
Total params: 2,128,193		
Trainable params: 2,128,193		
Non-trainable params: 0		

Fig. 12 Keras model summary

4.6.3 Neural Networks

Keras is a powerful and open-source Python library for development and evaluation of deep neural networks. It wraps the efficient numerical computation libraries for simple implementation [23]. For our neural network model, *Gender* Column is label-encoded and *Age*, *Gender*, *Stay_In_Current_City_Years* were encoded using One-Hot Encoding. The entire training data was scaled using Sklearn's MinMaxScaler, which essentially scales each value in the dataset within a range of [0,1]. Count variables were generated for each unique entry of *User_ID*, *Product_ID*, *Product_Category_1*, *Product_Category_2* and *Product_Category_3*. Additionally, Min, Max, Mean, 25_Percentile, 50_Percentile and 75_Percentile Variables are generated for *User_ID* and *Product_ID*. Lastly, Meanprice variables were generated for each of the three Product Categories. The structure of the neural network model is shown in Fig. 12. Adam Optimizer was used for optimizing the parameters of the model, with Learning_Rate set to 0.001 and using mean_squared_error as the loss function. Early_stopping [24] helped in stopping the training after 58 Epochs and restoring the best weights.

Table 10 AutoML’s leaderboard summary

MODEL_ID	RMSE
StackedEnsemble_AllModels_AutoML_20201228_143416	2389.45
StackedEnsemble_BestOfFamily_AutoML_20201228_143416	2397.41
GBM_5_AutoML_20201228_143416	2399.93
GBM_4_AutoML_20201228_143416	2401.47
GBM_grid__1_AutoML_20201228_143416_model_2	2408.11
GBM_3_AutoML_20201228_143416	2410.73
GBM_2_AutoML_20201228_143416	2416.09
GBM_1_AutoML_20201228_143416	2422.5
GBM_grid__1_AutoML_20201228_143416_model_1	2426.08
DeepLearning_1_AutoML_20201228_143416	2484.77
DeepLearning_grid__1_AutoML_20201228_143416_model_1	2558.13
GLM_1_AutoML_20201228_143416	4936.76

4.6.4 AutoML

H2O’s AutoML [25] is an extremely helpful library whose purpose is to provide a simple wrapper function to automate and simplify the machine learning workflow, by performing a large number of modelling-related tasks, for example - automatic tuning, training, and testing of multiple number of models, and this count is specified by the user. Previously trained models are used to create stacked ensembles, and these ensembles are further trained on clusters of individual models. The result is ensemble models which able to achieve high prediction scores. AutoML leaderboard can be printed at the end of training which presents the test metric scores of various models that it trained and tested. These models usually have different hyperparameters from each other, and stacked ensembles are most often the models with the least errors [26]. For our problem, *Age*, *Gender*, *Marital_Status*, *City_Category* & *Stay_In_Current_City_Years* were converted to string type for appropriate encoding by AutoML. Train and test data was converted to H2OFrame. A local H2O instance was setup on our personal device and 8GB RAM was allocated to it. The only parameter was `max_models`, which was set to 12. AutoML’s Leaderboard as displayed in Table 10 provides valuable information about each prediction model [26] and shows that AutoML tried various combinations of GBM – Gradient Boosting Machines, GLM – Generalized Linear Models, deep learning models, and lastly, two ensemble models. It can be concluded from this table that the stacked ensemble of all models was actually the best performing model.

4.6.5 Catboost

CatBoost [21] is a recently open-sourced machine learning algorithm (from Yandex) which has the capability of integrating with existing deep learning frameworks. Similar to XGBoost, CatBoost is built from the gradient boosting library, which

itself is a powerful machine learning algorithm, and is therefore applied to various types of problems in various domains. It is extremely robust as the data essentially needs little to no pre-processing at all. One implication is that it is possible to either specify categorical columns before training or to just leave them in string format itself.

The CatBoost library provides state-of-the-art results, on various different types of categories of data namely image, audio, text, etc. It uses a technique where there is tree-level weighted sampling, instead of split-level sampling. The main aim is to maximize the split-scoring accuracy [11]. This model has an argument named *cat_features* where one can specify the categorical features before training [21]. A list of categorical variables was prepared and passed to this argument. This list had the following entries – *User_ID*, *Stay_In_Current_City_Years*, *Product_ID*, *Age*, *Occupation*, *City_Category*, *Gender*, *Marital_Status*,. Numerous hyperparameters were tested using GridSearchCV method, and the most optimal values were found to be – *Learning_rate* = 0.04, *max_depth* = 10 and *iterations* = 4000.

Similar to the XGBoost model, the feature importance (Table 11) of the newly generated features is generally higher than that of the original features. Hence, it can be concluded that these data engineering techniques helped build a much more accurate model.

4.6.6 Ensembles

From the previously generated models, it is possible to make new models by taking weighted averages of old models. It was observed that the best score was obtained by taking the ratio of **57:43** weighted average of CatBoost and XGBoost models respectively. Adding any other model only degraded the performance (Table 12).

5 Performance Evaluation

This section discusses the configuration settings and experimental results.

5.1 Configuration Settings

The GPU-based high-performance computing environment is utilized to analyze the performance of various artificial intelligence and machine learning techniques for e-commerce applications. The entire experimentation has been conducted on a Windows 10 machine with the following specifications- Processor – Intel® Core i7-10750H, RAM – 16GB 3200MHz DDR4, GPU- Nvidia RTX 2060 (6GB VRAM) and System Type – 64 bits. The Python3 version is 3.6.3 and Anaconda Jupyter Notebook has been used for writing the python codes.

Table 11 Feature importance for Catboost model

Features	Importance (%)	Features	Importance (%)
PID_MeanP	12.875408	Pc2_MaxP	1.074566
PID_75Perc	6.35093	Product_Category_1_Count	1.022687
UID_25Perc	5.294287	Pc3_MinP	0.789584
PID_50Perc	5.03876	Product_Category_3_Count	0.787685
UID_MeanP	4.845355	Pc1_MaxP	0.748969
UID_75Perc	4.667984	Pc1_MeanP	0.729837
PID_25Perc	4.663049	Age_Count	0.727354
User_ID	4.597544	Pc3_MaxP	0.720994
UID_50Perc	3.98667	Pc1_50Perc	0.716456
UID_MaxP	3.732204	Gender	0.698784
User_ID_Count	3.188741	Pc1_75Perc	0.676638
UID_Min	2.962162	Pc1_25Perc	0.595421
Product_Category_1	2.742421	Occupation_Count	0.466051
Occupation	2.646217	Product_Category_2	0.400199
Purchasing_power	2.476389	Product_Category_3	0.376655
PID_MinP	2.45993	Marital_Status	0.293158
Product_ID	2.362142	Pc2_25Perc	0.242846
Product_ID_Count	2.2559	Pc3_25Perc	0.18234
Age	2.231589	Pc3_75Perc	0.179048
City_Category	1.78054	Pc3_50Perc	0.170634
PID_MaxP	1.630509	Pc3_MeanP	0.154758
Stay_In_Current_City_Years	1.49476	Pc2_50Perc	0.138978
Product_Category_2_Count	1.23495	Pc2_75Perc	0.126565
Pc1_MinP	1.210996	Pc2_MeanP	0.125618
Pc2_MinP	1.12474		

Table 12 RMSE score on test data

Approach	RMSE
Linear regression (baseline)	4623.9775
Linear regression (feature-engineered)	2610.3038
Neural-network	2534.5756
AutoML	2484.0706
XGBoost	2461.0996
Catboost	2458.1701
<i>Ensemble - (0.57*CatBoost) + (0.43*XGBoost)</i>	2447.3112

5.2 Results and Discussion

It can be observed from Table 12 that XGBoost & Catboost models made much more accurate predictions than the other models. It should also be noted that these models trained much faster than the others and required lesser RAM as well. Since the baseline linear regression model formulates a linear relationship between the

output and the various numerical inputs, the categorical features had to be label-encoded. In this process, these essential characteristics of such input dataset were not appropriately transmitted to the model and hence the highest RMSE score is observed. After the application of various feature engineering techniques, these categorical features are passed to the model and a much lower RMSE score was observed for the same. With artificial neural networks, a variety of models were tested by varying different parameters such as the number of hidden layers and neurons in each layer. Even though there was better input-output mapping when compared to linear regression, it was not as robust as the boosting algorithms used later in the experimentation. Also, with a large number of input features, training a neural network is computationally heavier and requires more memory and resources. We discuss ways to mitigate this in the future work.

Next, H2O's AutoML library works by training multiple models and formulating ensembles by trying various combinations. This library was able to produce slightly better RMSE scores but there is not much scope for fine-tuning hyperparameters since this is all done automatically. For problems with multiple features, it is observed that these tree-based boosting techniques were able to converge faster, whilst requiring lower computation compared to classic methods. Also, these libraries provided an extensive list of hyperparameters tuning which helped improve the model performance.

Finally, the ensemble-based boosting techniques such as the XGBoost and Catboost provided the best results. As of the date of writing this article, **the rank of our ensemble model is 119 out of a total of 23,159 submissions as shown in Fig. 13**. This proves the effectiveness and robustness of gradient-boosting algorithms, along with the necessity of feature engineering and augmentation.

6 Conclusion and Future Works

In this article, the dataset was thoroughly analysed, and it was found out that there was no direct correlation between any specific demographic and the Purchase price. Hence, new features were generated according to the importance of each existing feature. Then, the existing categorical data was appropriately converted into a numerical format as required by some algorithms. Extremely robust and modern data handling and prediction techniques were applied, along with the handling of outliers and erroneous predictions. All these techniques led to a respectable rank of 119 out of a total of 23,159 participants based on the RMSE on the public test data of the Black Friday contest.

6.1 Future Works

This work may be expanded on via the following ways:

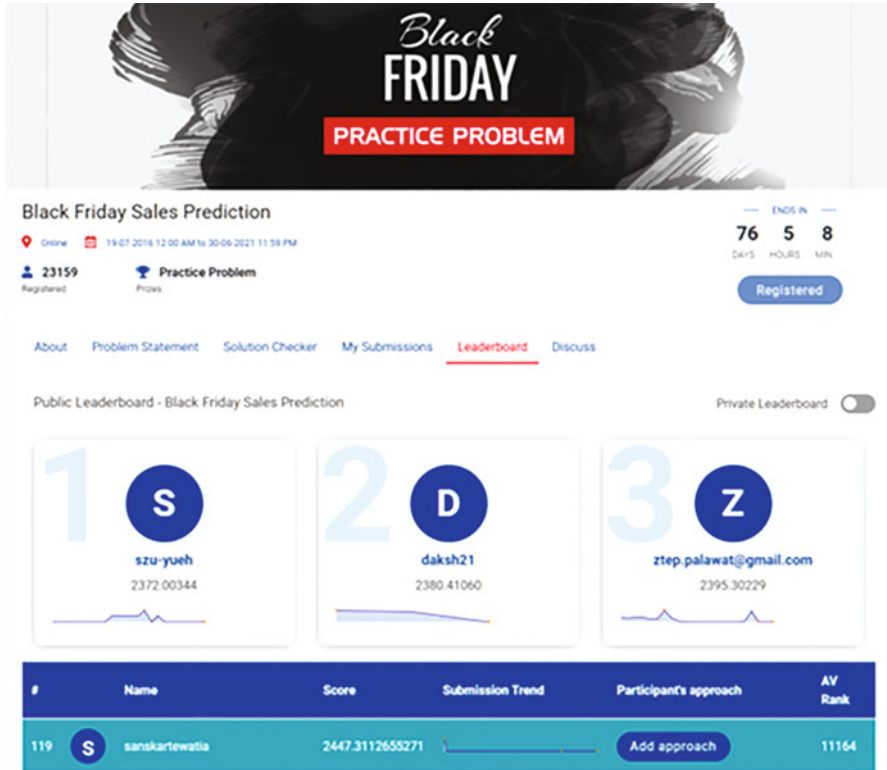


Fig. 13 The ranking of our submission to the Black Friday Sales Prediction contest

1. The data from the 2020 Black Friday Sales can be studied and used to make sales predictions in case of any other future pandemic.
2. A detailed dataset with more information about the actual selling products can help build a much better model.
3. The latest deep learning models can be utilized in the future to improve predictions [27].
4. Due to various factors like Covid-19 and the rise in Internet usage, shopping patterns have changed, so these machine-learning techniques can be applied to newly collected data from various e-commerce websites [7].
5. The exact same techniques of feature engineering and data augmentation can be applied to other problems (e.g., for healthcare, agriculture or weather forecasting) with limited data [27].

Next, as future directions, we give some insights into the advanced computing, distributed machine learning and privacy-preserving aspects of this or similar works.

6.1.1 Leveraging Advanced Computing

This line of work may be expanded upon by considering factors like scalability, security, and reliability, along with the implications such factors have on energy efficiency [28]. In future work, it is possible that an investigation of how security and privacy might be enhanced by utilising technologies based on blockchains and quantum computing [29]. This work has been applied to the e-Commerce sector using a case study of Purchase Prediction on Black Friday. Furthermore, this work can be applied to a variety of domains, including agriculture, healthcare, and smart homes, which are among the prospective application domains to explore. Last but not least, this research makes use of the Internet of Things (IoT) to capture real-time data rather than relying on datasets. The collected data is then deployed on a serverless edge computing environment in order to boost its efficiency and incorporate scalability [30]. Next, we also discuss the potential of expanding this work and similar approaches that involve large datasets based on leveraging distributed machine learning resources in the cloud or edge computing environments. And, we also shed light on the privacy-preservation aspects of work that involves dealing with users' data such as the case of the Black Friday Sale or similar datasets.

6.1.2 Scaling with Distributed Machine Learning

Distributed machine learning refers to techniques for training machine learning models on data that is distributed across multiple sources, such as computers or servers of retailers and organizations. There are several ways to apply distributed machine learning to a dataset like Black Friday sales data. One way to use distributed machine learning on this dataset would be to train a model on a distributed computing platform like Apache Spark or Horovod in the Cloud. This enables training the model using multiple machines, which can speed up the training process (esp. larger datasets). Another way to use distributed machine learning on this (or similar) dataset would be to train distributed models on different subsets of the data and then communicate the parameters, models or predictions from those models to make a final prediction. This is similar to ensemble learning, and it can be a powerful way to improve the performance of machine learning models. However, to mitigate the communication volumes fast techniques for gradient or model compression can be used [31]. And, to adapt to different network conditions, adaptive compression methods can help to improve the system performance [32]. Finally, there are many other techniques for accelerating distributed machine learning, and the best approach will depend on the specific needs and constraints of the project [33]. It is also worth noting that distributed machine learning can be more complex to set up and require specialized infrastructure, so it may not always be the best choice for every project.

6.1.3 Privacy-Preserving Machine Learning

Normally, one can encrypt the data before training the machine learning model on it to prevent anyone from gaining access to the model or the training data, however encryption cannot prevent the server from revealing the data. Privacy-preserving machine learning refers to techniques that allow you to train a machine learning model on a dataset without revealing the sensitive information to the model or to anyone else. Federated learning is one of the key approaches to apply this privacy preservation to Black Friday sales or similar datasets. FL allows training a machine learning model on multiple devices without sending the data to a central server. Instead, only the model parameters are sent back to the server to be aggregated. This way, the data stays on the devices and is never revealed to anyone. However, a key challenge with FL is the heterogeneity of the data, devices and users which can impact the model quality and bias [34]. To address this, approaches involving model quantization or pruning for mitigating the heterogeneity can be applied [35]. Moreover, it is possible to propose frameworks that combines several optimization methods to achieve resource efficiency [36]. It worth noting that federated learning also entails coordination and security concerns that require to be dealt with.

Software Availability

We have decided to make this experimental work on Black Friday Analysis and Prediction as open source. The python code, with descriptive plots, tables and results can be found at the following GitHub repository: <https://github.com/sanskartewatia/Sales-Prediction>

Acknowledgements This work is partially funded by Chinese Academy of Sciences President's International Fellowship Initiative (Grant No. 2023VTC0006), National Natural Science Foundation of China (No. 62102408), and Shenzhen Science and Technology Program (Grant No. RCBS20210609104609044).

References

1. Yi, D. (2010, November 23). Black Friday Deals for Target, H&M, Forever21, Old Navy, Radio Shack, and More. *Daily News*. New York. Archived from the original on August 15, 2011
2. Yahoo. (2010, November 23). Black Friday Moves to Thursday as Stores Woo Shoppers. Financially. *Yahoo! Finance*. Archived from the original on July 26, 2011. Retrieved January 2, 2012,
3. Chopra, M., Singh, S. K., Aggarwal, K., & Gupta, A. (2022). Predicting catastrophic events using machine learning models for natural language processing. In *Data Mining Approaches for Big Data and Sentiment Analysis in Social Media* (pp. 223–243). IGI Global.
4. Hossain, M. S., Uddin, M. K., Hossain, M. K., & Rahman, M. F. (2022). User sentiment analysis and review rating prediction for the blended learning platform app. In *Applying data science and learning analytics throughout a learner's lifespan* (pp. 113–132).
5. Peñalvo, F. J. G., Maan, T., Singh, S. K., Kumar, S., Arya, V., Chui, K. T., & Singh, G. P. (2022). Sustainable stock market prediction framework using machine learning models. *International Journal of Software Science and Computational Intelligence*, 14(1), 1–15.

6. Knowledge and Learning. (2016). Practice problem: Black Friday sales prediction | *Knowledge and Learning*, July 2016. [Online]. Available: <https://datahack.analyticsvidhya.com/contest/black-friday/>
7. Ifitikhar, S., Ahmad, M. M. M., et al. (2022). HunterPlus: AI based energy-efficient task scheduling for cloud-fog computing environments. *Internet of Things*, 21, 100667. (pp. 1–17). Elsevier.
8. Li, P., Li, D., Li, W., Gong, S., Fu, Y., & Hospedales, T. M. (2021). A simple feature augmentation for domain generalization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.
9. Jain, A.. (2021). *Blackfriday-AV*, URL: <https://www.kaggle.com/amanacden/blackfridayav/notebook>. Last Accessed on 27 May 2021.
10. Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794). Association for Computing Machinery.
11. Dorigush, A. V., Ershov, V., & Gulin, A., CatBoost: Gradient boosting with categorical features support. *arXiv preprint arXiv:1810.11363*. (2018)
12. Ifitikhar, S., et al. (2023). AI-based fog and edge computing: a systematic review, taxonomy and future directions. *Internet of Things*, 23, 100674. Elsevier.
13. Brdese, H. S., Alsaggaf, W., Aljohani, N., & Hassan, S. U. (2022). Predictive model using a machine learning approach for enhancing the retention rate of students at-risk. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 18(1), 1–21.
14. Basheer, S., Gandhi, U. D., Priyan, M. K., & Parthasarathy, P. (2022). Network support data analysis for fault identification using machine learning. In *Research anthology on machine learning techniques, methods, and applications* (pp. 586–595). IGI Global.
15. Barnston, A. G. (1992). Correspondence among the correlation, RMSE, and Heidke forecast verification measures; Refinement of the Heidke Score. *Weather Forecasting*, 7(4), 699–709.
16. Trung, N., Tan, D., & Huynh, H. (2019). *Black Friday Sale Prediction Via Extreme Gradient Boosted Trees*. [online] Available at: <http://vap.ac.vn/proceedingvap/proceeding/article/view/84>. Accessed 3 Jan 2023.
17. Kalra, S., Perumal, B., Yadav, S., & Narayanan, S. J. (2020). Analysing and predicting the purchases done on the day of Black Friday. In *International conference on emerging trends in information technology and engineering*. IEEE.
18. Xin, S., Ester, M., Bu, J., Yao, C., Li, Z., Zhou, X., et al. (2019). Multi-task based sales predictions for online promotions. In *28th ACM international conference on information and knowledge management*. Association for Computing Machinery.
19. Wu, C. M., Patil, P., & Gunaseelan, S. (2018a). Comparison of different machine learning algorithms for multiple regression on Black Friday sales data. In *IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE.
20. Ramasubbareddy, S., Srinivas, T. A. S., Govinda, K., & Swetha, E. (2021). Sales analysis on back friday using machine learning techniques. In *Intelligent system design: Proceedings of intelligent system design: INDIA 2019* (pp. 313–319). Springer Singapore.
21. Catboost. (2023). https://catboost.ai/en/docs/concepts/python-reference_catboostregressor, Accessed on 3 Jan 2023.
22. GridSearchCV. (2023). https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html. Accessed on 3 Jan 2023.
23. Keras Documentation. (2023). URL: <https://keras.io/api/>. Accessed on 3 Jan 2023.
24. Keras EarlyStopping. (2023). URL: https://keras.io/api/callbacks/early_stopping. Accessed on 3 Jan 2023.
25. H2O AutoML. (2023). URL: <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/automl.html>. Accessed on 3 Jan 2023.
26. He, X., Zhao, K., & Chu, X. (2019). AutoML: A survey of the state-of-the-art. *Knowledge-Based Systems*. arXiv preprint arXiv:1908.00709.

27. Gill, S. S., Xu, M., Ottaviani, C., Patros, P., Bahsoon, R., Shaghghi, A., et al. (2022a). AI for next generation computing: emerging trends and future directions. *Internet of Things*, 19, 100514.
28. Xu, M., Song, C., Wu, H., Gill, S. S., Ye, K., & Xu, C. (2022). esDNN: deep neural network based multivariate workload prediction in cloud computing environments. *ACM Transactions on Internet Technology*, 22(3), 1–24.
29. Gill, S. S., Kumar, A., Singh, H., Singh, M., Kaur, K., Usman, M., & Buyya, R. (2022b). Quantum computing: a taxonomy, systematic review and future directions. *Software Practice & Experience*, 52(1), 66–114.
30. Gill, S. S. (2021). Quantum and blockchain based Serverless edge computing: a vision, model, new trends and future directions. *Internet Technology Letters*, 24, e275.
31. Abdelmoniem, A. M., Elzanaty, A., Alouini, M.-S., & Canini, M. (2021). An efficient statistical-based gradient compression technique for distributed training systems. *Proceedings of the Machine Learning System (MLSys)*, 3, 297–322.
32. Abdelmoniem, A. M., & Canini, M. (2021a). Towards mitigating device heterogeneity in federated learning via adaptive model quantization. In *ACM EuroMLSys*. Association for Computing Machinery.
33. Xu, H., Ho, C.-Y., Abdelmoniem, A. M., Dutta, A., Bergou, E. H., Karatsenidis, K., Canini, M., & Kalnis, P. (2021). GRACE: A compressed communication framework for distributed machine learning. In *IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*. IEEE.
34. Abdelmoniem, A. M., Ho, C.-Y., Papageorgiou, P., & Canini, M. (2022). Empirical analysis of federated learning in heterogeneous environments. In *ACM EuroMLSys*. Association for Computing Machinery.
35. Abdelmoniem, A. M., & Canini, M. (2021b). DC2: Delay-aware compression control for distributed machine learning. In *IEEE conference on computer communications (INFOCOM)*. IEEE.
36. Abdelmoniem, A. M., Sahu, A. N., Canini, M., & Fahmy, S. A. (2023). Resource-efficient federated learning, *ACM EuroSys*. arxiv preprint arXiv:2111.01108.

Air Quality Index Prediction Using Various Machine Learning Algorithms



Mann Bajpai, Tarun Jain, Aditya Bhardwaj, Horesh Kumar,
and Rakesh Sharma

Abstract One of the most critical factors for human survival is air. The quality of air inhaled by humans affects their health and lives significantly. The continuously rising air pollution is a significant concern as it threatens human health and is an environmental issue in many Indian cities. A proper AQI prediction system will help tackle the problem of air pollution more efficiently and mitigate the health risks it causes. Government agencies use the Air Quality Index, a number to indicate the pollution level of the air to the public. It qualitatively illustrates the current state of the air. Aggregate values of PM_{2.5}, PM₁₀, CO₂, NO₂, and SO₂ have been taken to forecast the AQI for Pune city using the dataset collected by Pune Smart City Development Corporation Limited and IISc in 2019. This study aims to find the machine learning method which forecasts the most accurate AQI and its analysis.

Keywords Air quality index · Prediction · Machine learning · Linear regression · Random forest · KNN

1 Introduction

Over the past decade, the continuous rise in the air pollutants level in the atmosphere is one of the major emerging issues faced by the world. This problem has been fuelled by factors such as urbanization, industrialization, rapid growth in India's population and vehicles in the country, etc. The air pollutants mainly include nitrogen dioxide, carbon monoxide, ozone and sulfur dioxide, etc. the Air quality

M. Bajpai · T. Jain (✉) · R. Sharma
Manipal University Jaipur, Jaipur, Rajasthan, India

A. Bhardwaj
Bennett University, Greater Noida, Uttar Pradesh, India
e-mail: aditya.bhardwaj@bennett.edu.in

H. Kumar
Greater Noida Institute of Technology, Greater Noida, Uttar Pradesh, India

Fig. 1 AQI index range

Air Quality Index-Particulate Matter	
301–500	Hazardous
201–300	Very Unhealthy
151–200	Unhealthy
101–150	Unhealthy for Sensitive Groups
51–100	Moderate
0–50	Good

index (AQI) has been devised to get an idea of the concentration of pollutants in the air and their quality. It has been divided into 6 categories with specific colors and health hazard levels [1]. AQI ranges from 0–500, and a higher AQI value indicates greater levels of air pollution. From Fig. 1, it can be observed that for the range 0–50, the air quality is satisfactory; between 51 and 100, it’s acceptable but might be a risk for some people; in the range of 101–150 effects on health can be experienced by sensitive groups’ members. The range of AQI values 151–200 is considered unhealthy and can adversely affect human health. A content of 201–300 is considered very harmful, and AQI values higher than 300 are considered hazardous [2].

Pune has witnessed over 733 deaths per million people because of cardiovascular diseases developed due to exposure to (PM10 and SO₂) due to air pollution. People already suffering from lung diseases like pneumonia and asthma are more susceptible to lung and heart diseases on exposed to polluted air. The inhalation of air contaminated with PM2.5 and PM10 makes self-purifying the human immune system very difficult. Results from the emission inventory for PM2.5 of Pune city showed that half of the primary emissions come from the transport sector. Air pollution was also directly emitted from sources like industrial operations, resuspended dust, solid fuel combustion, etc. [3]. This paper aims to find the best air quality index prediction method for Pune city by implementing and evaluating various machine learning algorithms like regression, support vector machine, k-nearest neighbor algorithm, and random forest. In this project, firstly, the selection of significant and relevant features has been made, following which there has been the implementation of a different machine learning model for the prediction of the pollutants to yield highly accurate and error-free results for the AQI estimation.

The remainder of this paper is organized as follows. Section 2 presents a state-of-the-art existing related work. The working methodology of the proposed work is discussed in Sect. 3. Section 4 presents the performance evaluation parameters. Finally, results, followed by concluding remarks, are presented in Sects. 5 and 6.

2 Literature Review

Several studies and research have been done to predict the air quality index of different cities. These studies mainly focus on accurate estimation of the air quality index of cities for the formulation of better plans and preventive measures for controlling the air pollution levels in developing smart cities as well as the existing ones. The data collected by the sensors and actuators help understand the correlation between the different pollutants, the major pollutants causing severe damage to the human respiratory system, and the pattern in the emission of these pollutants.

In [4], Moolchand Sharma et al. presented a model to predict the AQI with an emphasis on performance and accuracy. Its robustness and accuracy were validated after testing six different machine learning classifiers. Different combinations of classifiers were tested to check which one gave the most accurate results. An accuracy of 99.7% was achieved using a Decision Tree, which increased by 0.02% when the Random tree classifier was applied. The study aimed to show the possibility of improvement in the AQI forecast using nonlinear machine learning algorithms.

In [5], Mehzabeen Mannan et al. have reviewed studies from different countries regarding the progress made in IAQ research, examining parameters like volatile matters, PM, carbon dioxides, and monoxides. Most works are focused on VOCs using gas chromatography-mass spectrometry for their analysis. Significant contributors to VOC concentrations are building structure and materials; for PM levels, it's the construction process and human movement and for indoor NH_3 it's concrete additives as per the research that has been reviewed in this work.

The pattern and air pollution trends have been analyzed for Delhi, Chennai, and Kolkata in [6] by Shrabanti Dutta et al. The air quality index has been developed using four major pollutants for 3 years. PM10 is the major pollutant affecting the air in all three cities. The climatic conditions are a significant factor in a place's air pollution along with the pollutant's seasonal distribution. The only pollutant accomplishing the NAAQ standard is SO_2 . The rest of the pollutants have emissions much higher than the NAAQ standards.

In [7], C. Amruthadevi et al. have compared different machine learning algorithms like Statistical multilevel regression, Neuro-Fuzzy, Deep Learning Long-Short-Term memory (DL-LSTM) and Non-Linear Artificial Neural Networks (ANN). Results show that the DL-LSTM is the most suitable algorithm for analyzing and forecasting pollutants in the air. Parameters used to compare the results include RMSE, MAPE and R^2 . In R^2 , a deviation in the range of 0.71–0.89 is there during the prediction of the pollutants' contamination level.

In [8] to forecast Wuhan city's air quality index, Al-Qaness et al. have proposed an adaptive neuro-fuzzy inference system. It has been named PSOSMA as it uses a slime mold algorithm (SMA), modified meta-heuristic algorithm (MH) and Particle Swarm Optimizer has been used to improve its performance (PSO). The data has been trained to predict the air pollutants like PM 2.5, SO_2 , CO_2 and NO_2 . The

performance of this proposed modified ANFIS, which uses PSOSMA is better than its counterparts.

R. Senthil Kumar et al. in [9], have proposed a method for analyzing and visualizing Bengaluru's AQI. Attribute selection methods like correlation matrix and decision tree have been used to analyze the important pollutants which are selected. The J48 decision tree has been used to select features with maximum gain ratio. Input data's similar features have been removed using Correlation matrix analysis. Data analysis and the calculation of results have been done using Expectation Maximization (EM) Clustering.

In [10], RM Fernando et al. predicted the concentration of PM_{2.5} in Columbo using the concentrations of air pollutants. The training and evaluation of the prediction model were done using machine learning algorithms like SVM, KNN, Random forest and Multiple Linear-Regression. The Random forest model had over 85% accuracy.

In [11], Ditsuhi Iskandaryan et al. studied the research works related to air quality prediction. The main observations were that most of the datasets being used, over 94.6%, were meteorological. At the same time, the rests were spatial and temporal data, and a large majority of the studies used open datasets too. To supplement the data gathered using air quality sensors, about 26 datasets have been used, which include 'Temporal', 'MET', 'Social media and 'Spatial' etc. The parameter of the analysis of the papers includes prediction target, type of dataset, data rate, algorithm, case studies, time granularity, etc. The authors found Random Forest, Support vector machine, and LSTM to be the most widely used methods for predicting particulate matter.

In [12] Kadir Diler Alemdar et al. have proposed a geographic information system-based approach for redesigning mitigation strategies in accordance with the risk classification and assumed scenario. The study aimed to demonstrate the changes in the mobility of traffic and the improvement in air quality due to the restrictions applied during the pandemic. It was observed that the level of air pollutants like PM₁₀, CO, SO₂, NO₂ etc. decreased significantly and the speed of traffic improved.

In [13], Laura Gladson et al. have developed an air quality index that shows the health risks caused by outdoor pollution in children. The creation of indices evaluated the impact of air pollutants like fine matter, ozone, nitrogen dioxide, etc. The indices presented normal distributions of locally scaled index values after adjustment and use values of the daily index of air pollutants. The author has provided the resources and steps for applying the final adjusted indices.

In [14], Xiali Sun et al. have proposed an IPSO – BP forecasting model which optimizes BP neural network's threshold and particle swarm weights. It's based on an improvised PSO-BP algorithm. The model improved prediction accuracy in comparison with BP and GA-BP. The particles search the optimal initial Value and BP's threshold value to create an IPSO-BP model for forecasting. This enhances the prediction accuracy and reduces the MAE too.

In [15], Narathep Phruksahiran et al. have proposed a geographically weighted predictor method for the hourly prediction of variables. The methodology combines GWP techniques and machine learning algorithms for the prediction of pollutants in the air on an hourly basis. It has better and more accurate forecast in all horizons compared to the existing prediction methods, improving the AQI prediction accuracy.

In [16], Manmeet Singh et al. analyzed the air quality across the globe using merged products of air pollutants and spatiotemporal resolution satellites during the COVID-19 lockdowns and found significant reductions in the concentrations of Nitrogen Dioxide, PM_{2.5}, and aerosol optical depth.

In [17], Subhashini Penetiet al. introduced blockchain-defined networks and a grey wolf-optimized modular neural network approach to managing intelligent environment security. User authentication-based blocks are designed for security in the construction, translation, and application layers. In IoT-enabled innovative applications, the maintenance of latency and computational resource utilization is done by applying optimized neural networks. In the results, the system ensures higher security and lower latency as compared to deep learning networks and multi-layer perceptron.

In [18], Dmitry Kochetkov et al. studied and discussed the implementation and development of 5G-based technologies for an urban environment. In the selected areas, a scientometric analysis of the field and a study of patent landscapes was conducted for the analysis of new technologies. The study of citation patterns was the object of scientometric analysis.

In [19], Ting Li et al. proposed a novel method for data collection from multiple sensor devices by partnering vehicles and unmanned aerial vehicles (UAV) in IoT. Using a genetic algorithm, vehicle collectors are selected for data collection through sensors following which collection routes of UAVs are planned using a novel deep reinforcement learning(DLR) based- route policy. Experiments conducted demonstrated that the proposed scheme reduces collection costs, and improves the coverage ratio of data collections for future 6G networks.

In [20], Aparna Kumari et al. presented a review of IoT and blockchain technology's functionality for smart cities. A blockchain-based decentralized architecture for IoT smart cities has been proposed which covers different application perspectives like Intelligent Transportation Systems, smart grids, and underlying 6G communication networks, giving directions to efficiently integrate blockchain into IoT-envisioned smart cities.

In [21], Metehan Guzel et al. have reviewed AQI prediction alliteration from an algorithmic view and have introduced a new air quality framework that processes large quantities of data in real-time using Complex Event Processing. Scalability and extendability are achieved using fog computing, and the manageability is enhanced using a software-defined network.

3 Methodology

Let's take a brief look at the description of the data. After the screening and analysis of the data, it is split into two parts; one is used for training and the other for testing. We'll be using different machine learning algorithms to maximize the accuracy of AQI prediction. The machine learning algorithms used in this project include SVM, Linear regression, Random Forest, Decision tree, XGBoost, and RNN.

3.1 Dataset

Descriptions of list of variables used is shown in Table 1.

PM2 has been used to represent the concentration of particulate matter with a size less than 2.5 microns and PM10 for matter with a size less than 10 microns. Similarly, SO₂, NO₂, CO, and O₃ have been used to represent sulfur dioxide, nitrogen dioxide, carbon monoxide and ozone respectively.

The data that we have worked with in this project is Smart City Testbed's subset which IISc Bangalore and Pune Smart City Development Corporation Limited collected in 2019 while using smart city testbed to solve simple to complex use cases. Provided the parameters and data for a particular region in Pune, this data can be used to predict the Air quality index of that particular area, for example, airports, IT hubs, Residential areas, Railway stations etc. The analysis of the Air quality index based on the data attributes present in this dataset like the percentage of different pollutants in the air, light, sound, etc., can significantly help improve the city's living conditions.

3.2 Data Collection and Pre-processing

The data set used in the study has been taken from Kaggle [22]. However, in this study, we have done a lot of data cleaning and preprocessing to remove the outliers and null values, etc. We have considered the averages of the maximum and

Table 1 Data variables and description

Variable	Description
PM2	Average of particulates <2.5 microns maximum and minimum
PM10	Average of particulates <10 microns max
SO ₂	Average of Sulphur dioxide maximum and minimum
NO ₂	Average of nitrogen dioxide maximum and minimum
CO	Average of carbon monoxide maximum and minimum
O ₃	Average of maximum and minimum

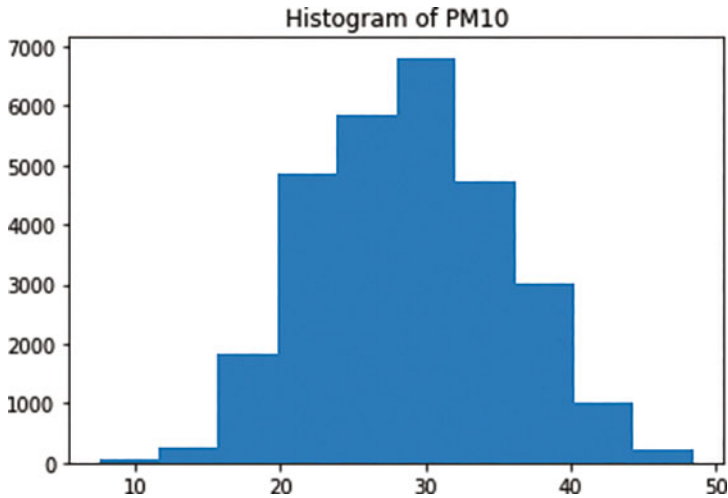


Fig. 2 PM10 histogram

minimum values of air pollutants like Ozone (O_3), Particulates <2.5 microns (PM2), Particulates <10 microns (PM10), Sulphur dioxide (SO_2), and Nitrogen Monoxide (NO).

3.2.1 Independent Variable Analysis

Histograms have been used to represent the relationship between independent variables and their frequency.

As shown in Fig. 2, we have a histogram for PM10. On the y-axis, we have the frequency and on the x-axis, we have the pollutant.

Figure 3 is the histogram for PM2. On the y-axis, we have the frequency; on the x-axis we have PM2.

Figure 4 is a histogram for NO_2 . On the y-axis we have the frequency; on the x-axis we have the pollutant.

In Fig. 5 we have a histogram for O_3 . On the y-axis, we have O_3 's frequency and on the x-axis, we have the pollutant.

Figure 6, we show a histogram for SO_2 . On the y-axis, we have the frequency and on the x-axis we have SO_2 .

Figure 7 is the histogram for CO. On the y-axis, we have CO's frequency and, on the x-axis, we have the pollutant.

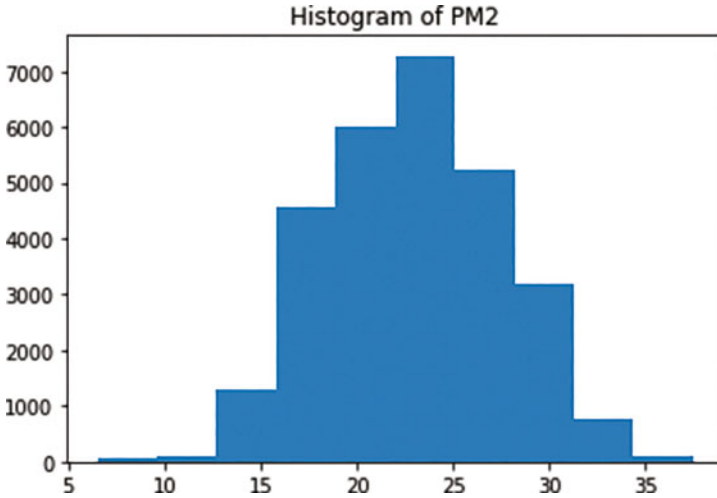


Fig. 3 PM2 histogram

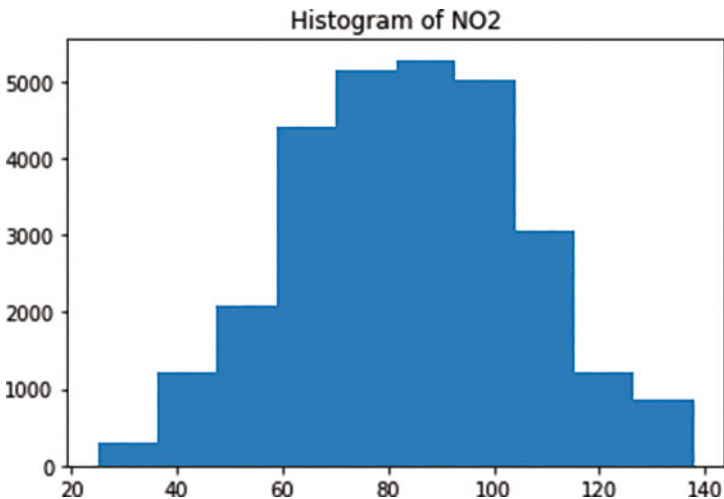


Fig. 4 NO₂ histogram

3.3 Data Analysis

Correlation matrix and distribution charts are used to determine the correlations among air pollution variables and the dataset's distribution and nature. For the analysis of data, Google collab notebooks have been used. In a dataset, correlations between all possible pairs of features are depicted using a Correlation matrix as shown in Table 2. The same has been used for the identification of features that

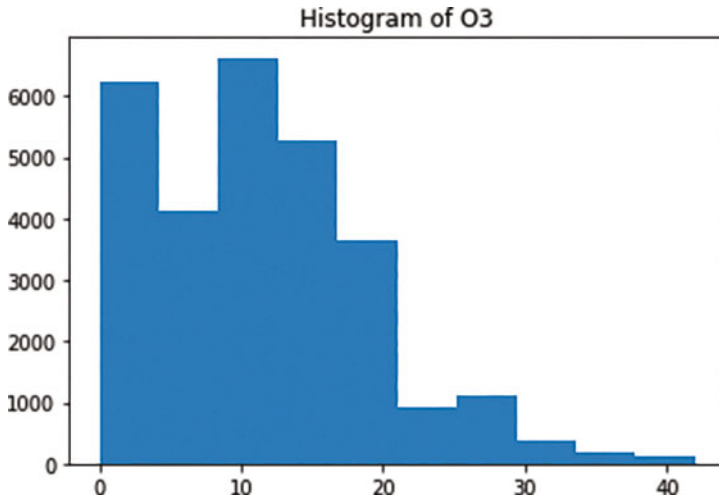


Fig. 5 O₃ histogram

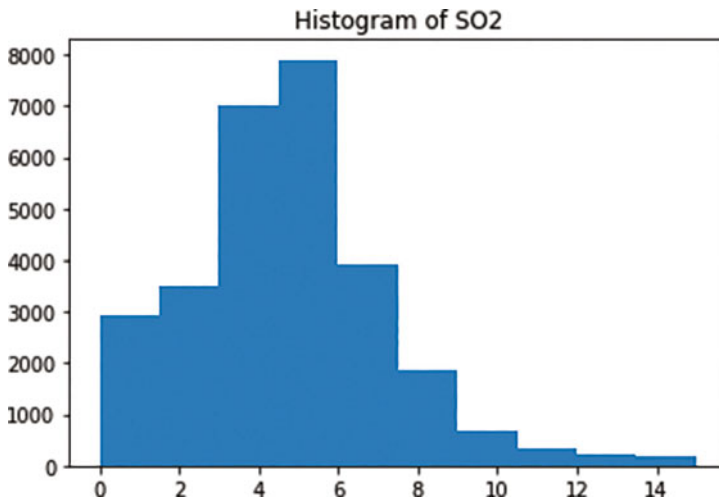


Fig. 6 SO₂ histogram

are most and least affected by PM2 to make the identification and visualization of patterns in the dataset easy and summarize it conveniently.

The above correlation matrix displays the correlation coefficient between pollutants such as PM10, PM2.5, SO₂, O₃, NO₂, and CO with each cell correlating the pollutants corresponding to the respective row and column. A 2D correlation matrix between two dimensions can be pictorially represented using a correlation heatmap, representing data with coloured cells from a monochromatic scale. Rows of the table are formed using values of the first dimension and the columns consist of values

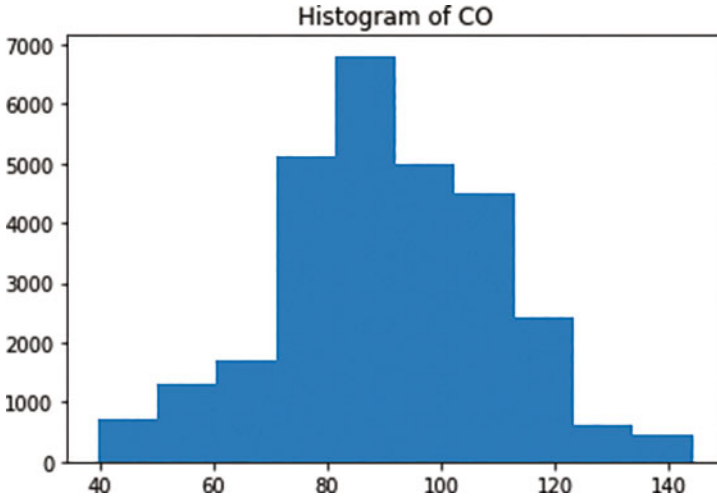


Fig. 7 CO histogram

Table 2 Correlation matrix

	Name	PM10	NO ₂	O ₃	PM2	SO ₂	CO
Name	1	-0.19626	0.222522	0.09315	-0.19547	0.121699	-0.4072
PM10	-0.19626	1	0.199159	0.090565	0.965509	-0.02489	0.408033
NO ₂	0.222522	0.199159	1	-0.19696	0.206393	0.195102	0.277064
O ₃	0.09315	0.090565	-0.19696	1	0.044566	0.208589	-0.13103
PM2	-0.19547	0.965509	0.206393	0.044566	1	-0.04153	0.468013
SO ₂	0.121699	-0.02489	0.195102	0.208589	-0.04153	1	0.000685
CO	-0.4072	0.408033	0.277064	-0.13103	0.468013	0.000685	v1

from the second dimension. The cell color is proportional to the measurements' number which matches the dimensional Value and is assisted by a color bar to make it understandable. It highlights the variation and differences in data making the patterns readable.

3.4 Machine Learning Algorithms

In this study, we are dealing with a regression problem where we are supposed to investigate the relationship between the independent feature variables and the dependent target variable to be able to make the prediction of the target attribute using the selected feature attributes. Training datasets are used to train the regression models with the values of the target variable and provided the feature attributes, the model learns to forecast the target variable [23, 24].

3.4.1 Support Vector Machine

It is one of the most commonly used machine learning algorithms and can be used for regression and classification's segregates n-dimensional space into classes using decision boundaries to simplify the categorization of new features. Hyperplanes are nothing but these best boundary lines. Hence datasets are divided into classes by SVM for finding a maximum marginal hyperplane. It chooses extreme data points called vectors for the creation of hyperplanes.

3.4.2 Random Forest Model

Random forest falls under supervised machine learning algorithms and is mainly used for Regression and Classification problems. For classification problems, it can tackle datasets with categorical variables and can deal with continuous variables too for problems related to regression. Decision trees are built by a Random forest algorithm taking their average for regression and majority vote in case of classification.

3.4.3 Linear Regression

It is a model used to depict the relationship between one dependent variable and one or multiple independent variables. In Simple Linear Regression, just a single dependent or explanatory variable is present. In this model, the summation of the distance between the predicted and actual Value of data is calculated and a line is chosen where this sum is minimum.

3.4.4 LSTM

It stands for Long Short-Term memory network. It is a type of recurrent neural network where the input of the current step is the output of the previous step; hence it can learn order dependencies in problems related to sequence prediction. Apart from single data points, like images, it can process the whole sequence of data as it has feedback connections.

3.4.5 Decision Tree

A decision tree is one of the most used methods for supervised learning. Decision trees split data sets on the basis of different conditions. It is used for both regression and classification tasks. Tree representation is used by the decision tree algorithm for problem-solving where leaf nodes represent class labels and internal nodes represent attributes Using decision trees, boolean functions can be represented on discrete attributes.

3.4.6 XGBoost

It is a machine learning algorithm based on gradient-boosted decision trees and has become very popular for structured or tabular data. It has been designed for speed and performance by using the dfs approach for tree pruning and parallelized tree building.

4 Evaluation Parameters and Implementation

For each of the pollutants, a sub-index is calculated based on their concentrations, health impacts, and their standards. The Value of the overall AQI is calculated and reflected by the worst sub-index. By using the help of medical experts, health impacts caused by these pollutants for various AQI categories have been suggested. For the pollutants that we have considered in the study, the AQI values are as follows:

Table 3 show the details about the category in which a pollutant lies for a certain concentration of its particles in the air. In this study, we have taken PM10 as the target variable on the basis of which we will be predicting the AQI for Pune City. Depending on the category in which the estimated Value of PM10 lies, the AQI for that specific area can be predicted. We have used machine learning models like SVM, Linear Regression, Random Forest, LSTM, Decision tree and XGBoost for the prediction of PM10's Value for the AQI calculation.

5 Results and Discussion

All the machine learning models used in the study, including the Support Vector Machine, Random forest model, Linear Regression, Decision Tree, LSTM and XGBoost, gave accuracies of over 90%. The Accuracies of the models are as follows. The exactness of the selected regression models is shown in Table 4.

Table 3 AQI values

AQI category (range)	PM2.5	PM10	NO ₂	O ₃	SO ₂	CO
Good (0–50)	0–30	0–50	0–40	0–50	0–40	0–1.0
Satisfactory (51–100)	31–60	51–100	42–80	51–100	41–80	1.1–2.0
Moderate (101–200)	61–90	101–250	81–180	101–168	81–380	2.1–10
Poor (201–300)	91–120	251–350	181–280	169–208	381–800	10–17
Severe (301–400)	121–250	351–430	281–400	209–748	801–1600	17–34
Hazardous (401+)	250+	430+	400+	748+	1600+	34+

Table 4 Models' accuracy

Model	Accuracy
Support vector machine	0.92307
Random forest	0.99964
Linear regression	0.93657
LSTM	0.991627
Decision tree	0.99958
XGBoost	0.97000

The accuracy of various different models has been displayed in the table above. Both the Random forest and Decision tree model had accuracy above 99.9%, but the Random forest beats the Decision tree's accuracy by 0.0001% hence for the given dataset Random Forest model is the best model. Moving on to the model evaluation metrics, we'll be calculating the Mean Absolute error, Mean Square Error, Root Mean Squared Error, and R-Squared Score for the models that have been selected for this study.

5.1 Model Evaluation Metrics

5.1.1 Mean Absolute Error (MAE)

It is an evaluation metric that measures the mean of the absolute difference between the actual and predicted values. Basically, it calculates the average of residuals in the dataset.

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (1)$$

Where N = Number of data samples,

\hat{y}_i = Predicted Value of y,

y = Actual Value of y.

5.1.2 Mean Square Error (MSE)

This evaluation metric calculates the mean of the squared difference between original and predicted values. It's used for the calculation of the variance of residuals in the dataset.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2)$$

Where N = Number of data samples,
 \hat{y}_i = Predicted Value of y ,
 y = Actual Value of y .

5.1.3 Root Mean Squared Error (RMSE)

It is an evaluation metric that calculates the square root value of MSE. It calculates the standard deviation of residuals in the dataset.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (3)$$

Where, N = Number of data samples,
 \hat{y}_i = Predicted Value of y ,
 y = Actual Value of y .

5.1.4 R-Squared Score (R^2)

The R^2 score or the Coefficient of determination is an evaluation metric used for the evaluation of a regression model. It's used for the calculation of variance in the predicted values of the dataset. The Value of R squared will always be less than one irrespective of the values.

$$R^2 = 1 - \frac{SS \text{ Regression}}{SS \text{ Total}} = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (4)$$

Where, N = Number of data samples,
 \hat{y}_i = Predicted Value of y ,
 y = Actual Value of y .

As per the evaluation metrics, we prefer lower values for MAES, MSE, and RMSE for relatively better performance. For the R^2 score, we prefer having larger values for better performance. Its Value usually lies between 0 and 1. A negative value for R^2 suggests that the chosen model doesn't follow the pattern and trend of data.

5.2 Model Analysis

The model having lower MAE, MSE, and RMSE values is considered to have a better performance as per the evaluation metrics and a model with higher R^2 scores is preferred over the ones with lower R^2 scores. The R^2 Value usually lies between 0 and 1. From Tables 5, 6, 7, and 8 we get the R^2 , MSE, MAE, and RMSE scores respectively.

From Table 5, it can be observed that the Random forest regressor, Decision tree regressor, and LSTM regressor have the best performance with a score of 0.99965, 0.99961, and 0.99117 respectively. The Linear regressor and XGBoost also show a great performance with a score of .9365 and .9700 respectively. SVM’s performance falls short in comparison to the other regressors but is still very good with a score of 0.87219.

Coming to the MSE scores, the Decision tree regressor and the Random Forest regressor have the lowest values of 0.0534 and 0.04503 respectively hence having the best performance. SVM has an MSE score of 16.523 and hence is the least suitable regressor as per the MSE metric. LSTM, XGBoost and Linear Regressor have MSE scores of 1.1407, 3.8779, and 8.1996 respectively.

Table 5 R^2 scores

Model	R^2 score
Linear regression	0.93657
Decision tree	0.99961
Random forest	0.99965
SVM	0.87219
XGBoost	0.97000
LSTM	0.99117

Table 6 MSE scores

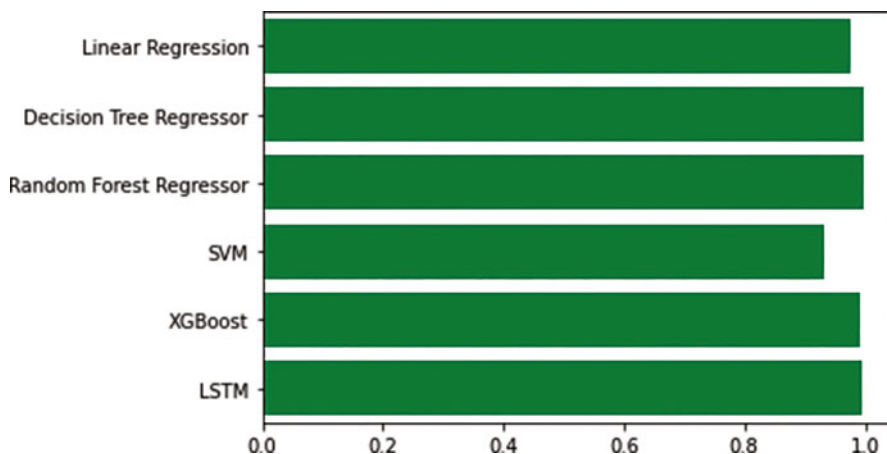
Model	MSE score
Linear regression	8.19964
Decision tree	0.05341
Random forest	0.04503
SVM	16.5235
XGBoost	3.8779
LSTM	1.1407

Table 7 MAE scores

Model	MAE score
Linear regression	1.76394
Decision tree	0.05521
Random forest	0.07546
SVM	3.04468
XGBoost	1.16371
LSTM	0.66770

Table 8 RMSE scores

Model	RMSE score
Linear regression	2.86350
Decision tree	0.23111
Random forest	0.21220
SVM	4.06491
XGBoost	1.96925
LSTM	1.06804

**Fig. 8** R² scores

For the MAE values Decision Tree and Random forest Regressor once again emerge as the most suitable options with MAE scores of 0.05 and 0.075 respectively. They are followed by LSTM, XGBoost, and Linear regressor and have MAE scores of 0.667, 1.1637, and 1.7639 respectively. SVM has the highest MAE score with a value of 3.044.

Following the trends of MSE scores, the RMSE scores of the Decision Tree and Random forest regressor have the best values of 0.2311 and 0.21220 respectively, followed by LSTM and XGBoost have a value of 1.068 and 1.969 respectively. Linear regressor and SVM have the highest RMSE values at 2.863 and 4.064 respectively and hence are the least suitable regressors.

5.2.1 Bar Plot of R²

Figure 8 displays the R² scores of the different models used in the study in the form of a bar graph. On the x-axis, we have the scores and on the y-axis, we have the different models.

Table 5 displays the R² scores of the different models used in the study in the form of table.

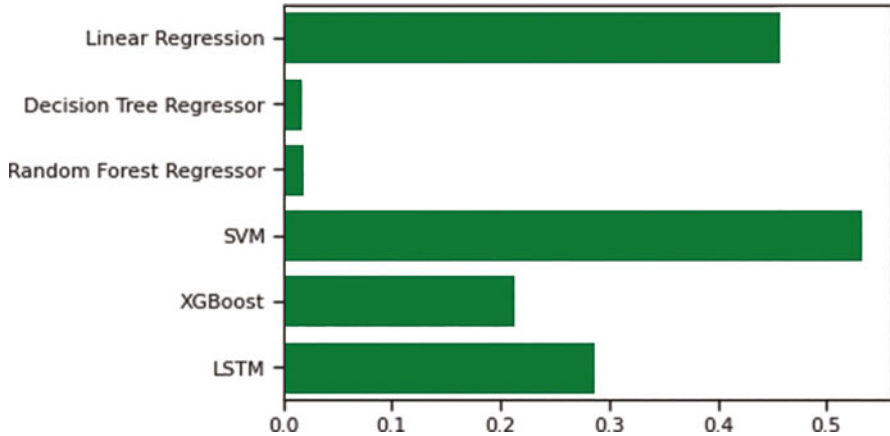


Fig. 9 MSE scores

5.2.2 Bar Plot of MSE

Figure 9 displays the MSE scores of the different models used in the study in the form of a bar graph. On the x-axis, we have the scores and on the y-axis, we have the different models used in the study.

MSE Scores

Table 6 displays the MSE scores of the different models used in the study in tabular format.

5.2.3 Bar Plot of MAE

Figure 10 displays the MAE scores of the different models used in the study in the form of a bar graph. We have the different models on the y-axis and the scores on the x-axis.

Table 7 displays the MAE scores of the different models used in the study in the form of a table.

5.2.4 Bar Plot of RMSE

Figure 11 displays the RMSE scores of the different models used in the study in the form of a bar graph. On the x-axis, we have the scores and on the y-axis, we have the different models which have been used in the study.

Table 8 displays the RMSE scores of the different models used in the study in tabular format.

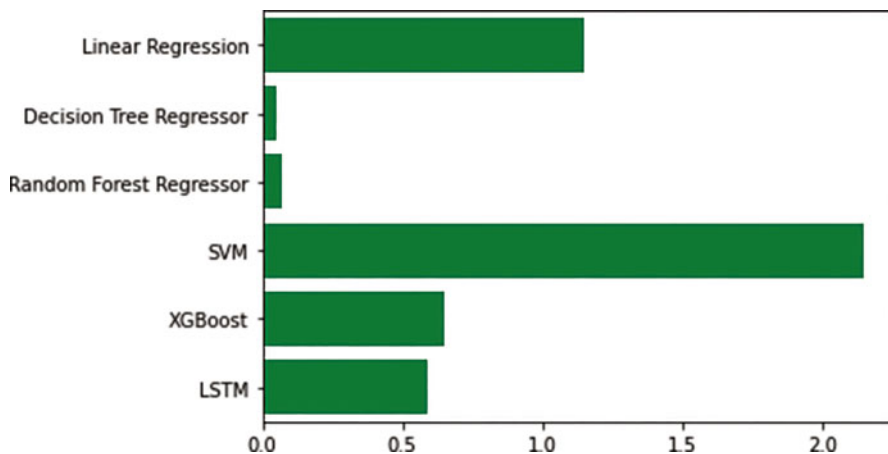


Fig. 10 MAE scores

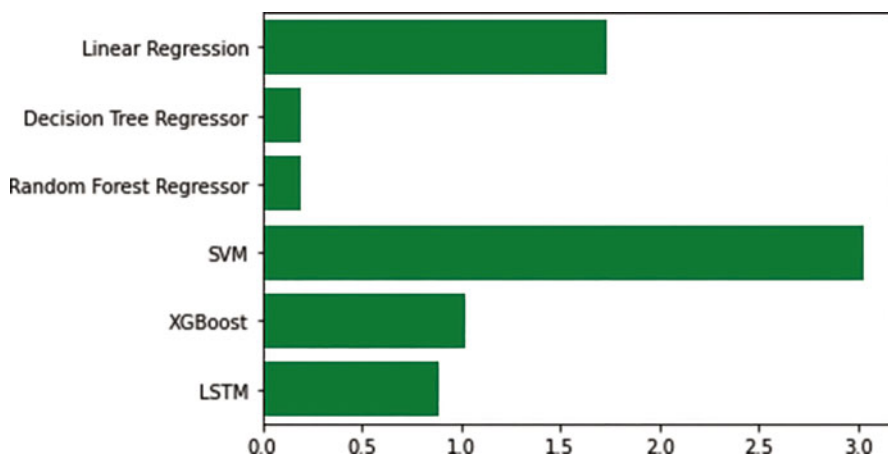


Fig. 11 RMSE scores

6 Conclusion and Future Scope

Air is a crucial element for human survival. Air Quality Index (AQI) value is a numerical representation of the current air quality. A correlation analysis was performed in this investigation to identify the contaminants influencing the air quality index. Pune's concentration of PM_{2.5} is anticipated using a rigorous correlation analysis. The current study assessed the accuracy of various deep learning and machine learning classification models on the dataset provided to estimate Pune's Air Quality Index (AQI). The dataset was preprocessed and cross-validated to increase prediction accuracy, with 70% of the data used for

model training and 30% for model testing. Support Vector Machine, Random Forest, Linear Regression, Decision Tree, LSTM, and XGBoost are included in the study's model. SVM obtained 92.307% accuracy, Random Forest 99.96%, Linear Regression 93.96%, LSTM 99.16%, Decision Tree 99.95%, and XGBoost model 97.0%. After evaluating all models with the most accurate predictions, the random forest emerged as the top model. Nonetheless, the Decision Tree model was also almost 99% more accurate.

Upon computing evaluation measures such as the R^2 score, MAE, MSE, and RMSE, it was determined that the Random Forest regressor had the greatest R^2 Value and the lowest MAE, MSE, and RMSE values, followed by the Decision Tree Regressor. The LSTM and XGBoost regressors also performed well on the dataset. The linear regressor also performed admirably. However, the SVM regressor had the lowest R^2 score and the highest MAE, MSE, and RMSE values for the dataset, indicating that it was the least acceptable model among the six investigated in this study. Future research could study the development of a prediction model based on deep learning that can determine the AQI of a given city or district.

References

1. WHO air pollution report. Available at: <https://www.who.int/health-topics/air-pollution>. Accessed 20 Sept 2022.
2. Air quality data statistics. Available at: <https://www.airnow.gov/>. Accessed 1 Oct 2022.
3. Central pollution control board. Available at: <https://cpcb.nic.in>. Accessed 5 Oct 2022.
4. Sharma, M., Jain, S., Mittal, S., & Sheikh, T. H. (2021). Forecasting and prediction of air pollutants concentrate using machine learning techniques: The case of India. In *IOP conference series: Materials science and engineering* (Vol. 1022, No. 1, p. 012123). IOP Publishing.
5. Mannan, M., & Al-Ghamdi, S. G. (2021). Indoor air quality in buildings: A comprehensive review on the factors influencing air pollution in a residential and commercial structure. *International Journal of Environmental Research and Public Health*, 18(6), 3276.
6. Dutta, S., Ghosh, S., & Dinda, S. (2021). Urban air-quality assessment and inferring the association between different factors: A comparative study among Delhi, Kolkata and Chennai megacity of India. *Aerosol Science and Engineering*, 5(1), 93–111.
7. Amuthadevi, C., Vijayan, D. S., & Ramachandran, V. (2021). Development of air quality monitoring (AQM) models using different machine learning approaches. *Journal of Ambient Intelligence and Humanized Computing*, 1–13.
8. Al-Qaness, M. A., Fan, H., Ewees, A. A., Yousri, D., & Abd Elaziz, M. (2021). Improved ANFIS model for forecasting Wuhan City air quality and analysis COVID-19 lockdown impacts on air quality. *Environmental Research*, 194, 110607.
9. Kumar, R. S., Arulanandham, A., & Arumugam, S. (2021, October). Air quality index analysis of Bengaluru city air pollutants using expectation maximization clustering. In *2021 international conference on advancements in electrical, electronics, communication, computing and automation (ICAECA)* (pp. 1–4). IEEE.
10. Fernando, R. M., Ilmini, W. M. K. S., & Vidanagama, D. U. (2022). Prediction of air quality index in Colombo.
11. Iskandaryan, D., Ramos, F., & Trilles, S. (2021). Features exploration from datasets vision in the air quality prediction domain. *Atmosphere*, 12(3), 312.
12. Alemdar, K. D., Kaya, Ö., Canale, A., Çodur, M. Y., & Campisi, T. (2021). Evaluation of air quality index by spatial analysis depending on vehicle traffic during the COVID-19 outbreak in Turkey. *Energies*, 14(18), 5729.

13. Gladson, L. A., Cromar, K. R., Ghazipura, M., Knowland, K. E., Keller, C. A., & Duncan, B. (2022). Communicating respiratory health risk among children using a global air quality index. *Environment International*, *159*, 107023.
14. Sun, X., Li, S., Chen, X., & Wang, K. (2021, March). Air quality index prediction based on improved PSO-BP. In *IOP conference series: Earth and environmental science* (Vol. 692, No. 3, p. 032069). IOP Publishing.
15. Phruksahiran, N. (2021). Improvement of air quality index prediction using geographically weighted predictor methodology. *Urban Climate*, *38*, 100890.
16. Singh, M., Singh, B. B., Singh, R., Upendra, B., Kaur, R., Gill, S. S., & Biswas, M. S. (2021). Quantifying COVID-19-enforced global changes in atmospheric pollutants using cloud computing-based remote sensing. *Remote Sensing Applications: Society and Environment*, *22*, 100489.
17. Peneti, S., et al. (2021). BDN-GWMNN: Internet of things (IoT) enabled secure smart city applications. *Wireless Personal Communications*, *119*(3), 2469–2485.
18. Kochetkov, D., Vuković, D., Sadekov, N., & Levkiv, H. (2019). Smart cities and 5G networks: An emerging technological area? *Journal of the Geographical Institute "Jovan Cvijić" SASA*, *69*(3), 289–295.
19. Li, T., et al. (2021). DRLR: A deep-reinforcement-learning-based recruitment scheme for massive data collections in 6G-based IoT networks. *IEEE Internet of Things Journal*, *9*(16), 14595–14609.
20. Kumari, A., Gupta, R., & Tanwar, S. (2021). Amalgamation of blockchain and IoT for smart cities underlying 6G communication: A comprehensive review. *Computer Communications*, *172*, 102–118.
21. Guzel, M., & Ozdemir, S. (2019). A new CEP-based air quality prediction framework for fog based IoT. *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE.
22. Pune smart city dataset. Available at: <https://www.kaggle.com/datasets/akshman/pune-smartcity-test-dataset>. Accessed 1 Oct 2022.
23. Soni, K. M., Gupta, A., & Jain, T. (2021). Supervised machine learning approaches for breast cancer classification and a high-performance recurrent neural network. In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 1–7. <https://doi.org/10.1109/ICIRCA51532.2021.9544630>aset easy and summariz
24. Jain, T., Verma, V. K., Agarwal, M., Yadav, A. & Jain, A. (2020). Supervised machine learning approach for the prediction of breast cancer. In *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, pp. 1–6. <https://doi.org/10.1109/ICSCAN49426.2020.9262403>

Leveraging Cloud-Native Microservices Architecture for High Performance Real-Time Intra-Day Trading: A Tutorial



Mousumi Hota, Ahmed M. Abdelmoniem, Minxian Xu, and Sukhpal Singh Gill 

Abstract Day trading has been gaining attention from prospective investors over the past decades, even more so in the last decade due to a plethora of factors such as instantaneous availability and accessibility to information such as social media, news, Internet of Things (IoT), availability of market's sentiment data associated with them, and increased broker discounts. This tutorial aims at providing a framework for intra-day trading that supports scalability, easily maintainable by creating a low coupling, high cohesion, and stateless architecture between client and server, considering the time-sensitive nature of transactions involved. This provides the benefits of additional business value for Software as a Service (SaaS) providers based on high productivity as well as enhanced end-user experience. To achieve these objectives, a combination of cloud-native architectural components, such as microservices and event streaming using Kafka, is used in this tutorial to provide a near real-time experience to end users. Additionally, to ensure security, robust authentication management is used in the proposed solution to control the access of read and write operations on the Firebase cloud database.

Keywords Cloud computing · Cloud-native architecture · Distributed systems · Microservices · Real-time stock monitoring

1 Introduction

Over the last decades, intra-day trading has evolved from a topic of interest within the regular trader's community to pre-investors, such as friends and close contacts,

M. Hota · A. M. Abdelmoniem · S. S. Gill (✉)
School of Electronic Engineering and Computer Science, Queen Mary University of London,
London, UK
e-mail: ahmed.sayed@qmul.ac.uk; s.s.gill@qmul.ac.uk

M. Xu
Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
e-mail: mx.xu@siat.ac.cn

who are not actively involved but started taking an interest in trading. Due to the availability of Software as a Service (SaaS) start-ups, intra-day trading has further piqued their interest. Since the art of intra-day trading can only be learned through observation and experience [1], when it comes to day trading, prospective investors need to focus their attention on a magnitude of trading metrics all at once in order to take holistic and pragmatic decisions regarding their investment. However, users are apprehensive about using such platforms due to concerns such as security, overall web application performance, and ease of use. On the other hand, intra-day trading SaaS providers face challenges such as scalability, cost-effectiveness, and high availability because of the high frequency of transactions involved.

The tutorial aims at presenting high-performance real-time stock data monitoring on a dashboard to support end users who are interested in day trading [21]. Momentum-based strategies were adopted as stock price changes (or volatility) could be magnified and hence could help users make better stock order decisions [1]. For enabling end users to have a 360-degree view of 5 leading and 3 lagging indicators, to assist them in making the assessment of risks and profits associated over an interval of 1 min (as long as the connection persists). The implementation of the project considers criteria such as Kafka event streaming, which offers low latency, flexibility, high throughput and failover (by considering multiple Kafka brokers) [2]. On the server side, it considers server-sent events to provide end users with the best experience of the consolidated stock chart view, updated every minute without any delay (with the exception of the delay on intervals mentioned).

1.1 Main Contributions

The main contributions of this work are:

- Proposed a framework cloud-native architectural framework for web applications using techniques such as microservices, Firebase for the backend, Kafka for middleware, and WebSocket server for establishing communication in a publish, subscribe manner.
- Designed microservices based approach to fetch stock data from Finhubb stock API using Asynchronous Promises.
- Built highly secure user management using Firebase authentication service.
- Optimised Enabled event streaming Kafka consumer by availing multi-broker support.
- Optimized user's order processing using Firestore for easier, fast access to backend user's data.
- Implemented the proposed architecture using web application run time environment as nodejs on both client and Server side.
- Presented detailed analysis of momentum trading strategy using leading indicators.
- Proposed future directions that could bolster the cloud offerings SaaS providers.

1.2 *Article Organisation*

The rest of the tutorial is structured as follows: Section 2 presents related work. Section 3 gives details about the technologies used. Section 4 provides broader insight into application architecture. Section 5 comprises implementation considerations. Section 6 concludes the proposed solution and highlights future directions.

2 **Related Work**

We have identified only one similar work in the literature. In [24] authors has adopted Kafka for event streaming by providing supporting evidence that increasing large event streaming leads to high processing time [24]. The authors have further provided evidence on how Kafka could make event streaming fault-tolerant, fast and secure. However, the author's work differs from the proposed solution in this paper, as it focuses on data analysis aspects of tweet data. In [24], authors have delineated how we can control access to stock data, that is, whether to share or hide required services using the Microservices approach. This could be very useful for scaling applications in future. Literature on cloud-native architecture reported [4–7] that this work is not been applied to investigate intra-day trading.

3 **Technologies Used**

This section introduces the details about the technologies used in this tutorial.

3.1 *Kafka*

Kafka provides continuous processing of events sequentially. Since consumer producers are decoupled, both can evolve and fail independently of the topic/Stream. Ordered records are immutable and can only add at the end. Records numbering usually starts from 0 and monotonically increases.

3.2 *Kafka Message*

Every event is a key (domain object that you can serialise and store there, key is integer/string), value pair. Every message has a timestamp, is automatically provided, and can be set explicitly.

3.3 *Broker*

It manages partitions or takes requests from updates from producers, and update them on partitions. Take requests from a consumer and writes them out. Storage and pub-sub Schema of the topic-retention period, compaction properties. Brokers need to agree on a consensus, zookeeper ensures:

- Leader Election
- Access control list
- Replication organization
- Failover

Broker ensures:

- Partition Rebalancing: The Kafka consumer group performs rebalancing of workload when a new partition is added
- Parallelization: With parallelization, multiple consumers can access data across multiple systems
- Durability: It is ensured by Kafka replication [8]

3.4 *Partition*

Each partition has a considerable amount of replicas. Generally, the replication factor is 3; one of them is a leader, and the others are followers. When you produce a partition, we actually produce it for the leader. The producer is connecting to the broker that has the lead partition there. The job of the brokers with follower partitions is to reach out to leaders and scrape the new messages and keep them up to date with them ASAP. Language support:

- Java native library
- Kotlin, Closure, Groovy
- Failover
- Supportive languages: Python, C, C++ (library Kafka)

3.5 *Producer*

Producer applies round-robin scheduling for writing to a topic. In reality, it actually writes to a partition. Partitions are even; a lot of orders in not in place in the round-robin mechanism. If the order is important, they need to be ordered by a key; the producer can hash that key and mod that key with the number of partitions, given the partition number that the producer is going to write to. The same key is always going to write to the same partition as long as the number of partitions is constant (usually the case).

3.6 *Leading Indicators*

Leading indicators are used to predict a trend before the phenomenon has happened and hence help traders/prospective traders make assessments of risks and benefits of their investment in advance. We discuss the key leading stock data indicators used in our application as below:

OHLCV—Open, High, Low, Close, Volume Represents Open, High, Low, and Close, Volume of a ticker every 1 min interval from Finnhub API.

RSI—“Relative Strength Index” [8]

- A technical indicator that shows traders the magnitude in which the price of stock changes. In general, conception, when the value is above 70%, shows that the stock is oversold, vs when the value is below 30, the stock is undersold.
- This momentum indicator is considered useful as the rising market(bull market) persists more than the bear market.

Stoch—“Stochastic Oscillator” [9]

1. This momentum indicator is based on the movement of the close price of a stock with a range of prices.
2. It relies on the price history
3. Its sensitivity can be smoothed using the moving average

ADX—“Average Directional Strength”

1. It is used to indicate the strength price trending, whether the price is going up, upward trend or moving down, downward trend.
2. Its trend strength by ADX can be measured as: if between 0 and 25 then weak, then the trend is strong, 50–75 shows very strong, and above 75 extremely strong.

OBV—“On-Balance Volume” [10]

1. This momentum indicator is based on changes in volume.
2. It demonstrates crowd sentiment for predicting whether the result is rising or falling.
3. The calculation is based on the principle that if consecutive close prices are the same, then obv is 0, if the current close is more than the previous close, then the volume is considered positive else, negative.

CCI—“Commodity Channel Index” [22]

1. This momentum indicator indicates the trend of when a stock is about be oversold or stock is over-brought
2. If overbought the price is corrected by market soon, indicating user it is not the right time to sell the stock. If oversold there is potential for the stock price to rebound soon, hence user can wait for that time to buy the stock.
3. If CCI rises from negative value to a value above 100, it shows uptrend.

3.7 *Lagging Indicators*

Lagging indicators are based on historical data and are used to identify patterns in data. Below are examples of Lagging indicators.

SMA—Simple Moving Average [23]

1. SMA calculation is based on average of closing prices(generally) from a selected range of closing prices.
2. Indicates if price would rise in future or not.
3. It smooths out price volatility to make it easier for traders to observe and analyse the price trend.

EMA—Exponential Moving Average [11]

1. It is based on selected range of prices, where more weightage is given to price data that is more recent.
2. It indicates points in chart where intersection price of security and EMA (in this case) occurs to signal buy, sell.
3. It is more efficient indicator than SMA.

TEMA—Triple Exponential Moving Average [12]

1. It is a technical indicator.
2. It follows same calculation as EMA with an exception that lags are subtracted, to capture quick changes in the price.
3. Its trend direction can be identified using TEMA.
4. Is more efficient indicator than EMA.

In order to have a full view of a company, other microservices were included:

```
get_FINNHUB_Quote ,
get_FINNHUB_Company_Profile ,
get_FINNHUB_Income_Details ,
get_FINNHUB_performance ,
get_FINNHUB_peers ,
get_FINNHUB_topnews ,
get_FINNHUB_recommedation ,
```

3.8 *Microservices*

With the growing size of systems, scalability becomes an issue. Using “Cloud-native architecture” such as microservices helps overcome this issue as they could be easily understood and scaled without dependency on each other [13]. In terms of software

development, it supports continuous delivery [14]. It supports scalability, reusability, resiliency, and cost-effectiveness. This can be supporting by evidence author in [3].

Microservices are created to read data from **Finnhub stock API** and were processed to useful JSON format for making it suitable for data consumption on the client side. Microservices were implemented using **Nodejs** using respective npm packages for the APIs indicated earlier. The following are details on how to use the microservices:

- No state or session variables are shared between the client and server.
- Each microservice is implemented as promise calls to the stock API endpoints within an asynchronous function using nodejs in two javascript files. The data returned was then processed in a required format and exposed to Kafka producer. In this approach caller (producer) starts producing messages when the data has been received from the microservices modules, that is, when the API end point promise calls were resolved. In the event of errors, for instance, if API call limit was reached per minute, error handled was also implemented in the microservices module.
- For real-time event streaming, from a performance perspective, Kafka was found to be a suitable technology for middleware.
- Having multiple brokers ensures durability; hence based on the requirements and scope of the project, the number of Kafka brokers chosen was three. Further, a centralized service was required to manage Kafka brokers within such distributed system setup to ensure robust synchronization, maintenance of metadata, and list of topics and messages produced and consumed. This was facilitated by **zookeeper**.
- The simplest way to install the aforementioned for setting up the environment for using Kafka was using a docker-compose command, file extension, and .yml. as the installation required multiple containers. A docker-compose YAML file was created with all the required configuration details including port numbers for respective Kafka, and zookeeper servers. As per the config file used, the Kafka broker used was named `kafka_broker_1`, `kafka_broker_2`, `kafka_broker_3`, and the zookeeper container was named `zookeeper`.

3.9 *Producer-Consumer*

- Kafka producer-consumer were implemented using `kafkajs`: [kafkajs producer](#), [consumer](#): [kafkajs consumer](#) [kafkajs npm package link](#)
- Kafka producer was used for writing to a topic on a specific partition so that in case of partition failure, topics written on other partitions were still available. Using Kafka subscribers, the written topics were subscribed to and consumed by Kafka consumers.

Observation When partitions were increased and set to 10 from 2, during testing, the consumer hanged and so took more time to start consuming topics.

3.10 Earlier Considerations

The chosen interval for the topic message creation is 1 min. Hence the producer needs to push the messages to Kafka topics and then be consumed by Kafka consumers on the server side. As soon as the client requests for the stock data to be rendered on the front end. Hence **sse**, server-sent events server-side events. On the client end, **EventSource API** methods are used to handle the message received from the consumer, which acts as a server in the given context.

4 Application Architecture

The application architecture developed in this work is shown in Fig. 1.

The architecture consists of Kafka service that manages the producers and consumers. The zookeeper service which manages the coordination among the various components in the application. The database service which stores and maintains the data. The APIs that are used to communicate with the client and apply the trade instructions. In the following, we discuss the details of the application architecture which is based on the Client-Server Communication paradigm. Specifically, we

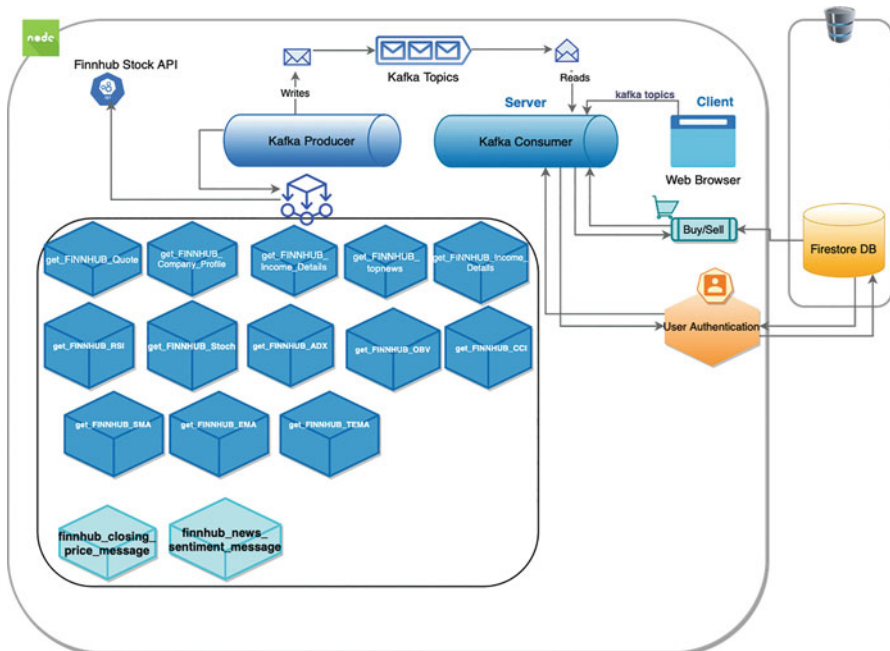


Fig. 1 Application architecture

analyze and select which method is most suitable for client Server Communication. We depict the client server communication options in this architecture from existing work [15]. It shows short polling, long polling and server-based events types of communication that are possible.

5 Implementation Highlights

- For any given day, the Finnhub stock API gives access to stock data available from 12:00 am to 9:00 am, weekends (Saturday, Sunday), hence date input is adjusted to the previous date for uninterrupted user experience.
- As Finnhub stock API used UNIX-based timestamps to fetch data from the API. Hence current date has been converted to human readable form (with suffixes AM or PM) and from Unix timestamps.

5.1 Brief Introduction to the API

- Finhub stock API that sources its data from multiple stock exchanges.
- Keys are obtained for gaining authorized access to the API endpoints.

5.2 Technology Stacks

Next, we describe the main technologies used in the development of the application.

Nodejs Nodejs is a cross-platform and open-source server environment that can run on Windows, Linux, Unix, macOS as well as other operating systems. It is used as a back-end JavaScript runtime environment to run on the JavaScript Engine, and execute JavaScript code outside a web browser.

Backend As the data are obtained and some amount of latency is expected, the below mentioned methods are implemented which promises better error handling as compared to its contemporary approaches.

Cloud Database Firebase provides us with the needed hosting services. It can host any application type (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++, etc.). The benefits is that it can run NoSQL databases and services like a real-time communication server and provide real-time hosting of content, social interactions (e.g., Twitter, Google, Facebook, and Github), and notifications.

Middleware Docker was used for building, deploying and managing containers. Docker containers are used to dockerize the various services used in the architecture

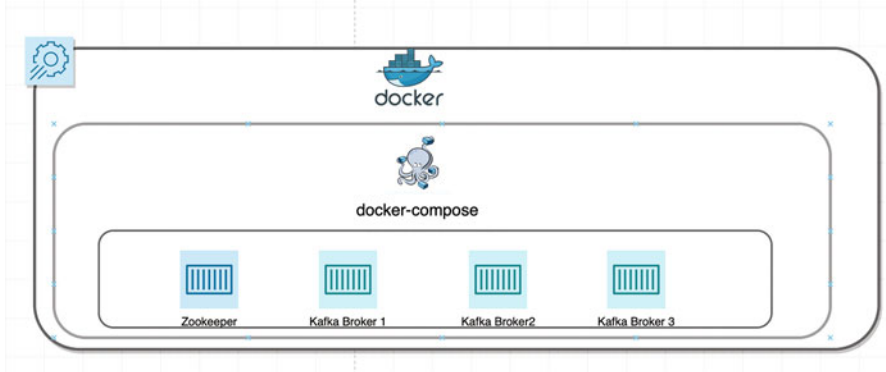


Fig. 2 Dockerizing Zookeeper, Kafka, and KSQLDB servers

such as Zookeeper, Kafka and KSQLDB servers as shown in Fig. 2. The containers required in the application implementation are as follows:

- Three **Kafka brokers** to enhance resiliency in case of any failovers, one container was build and deployed for each broker Note: There containers were configured to enable kafka producer and consumer to be able connect and write to, read from an available, elected leader broker from the pool of deployed brokers.
- One container was configured **zookeeper** for managing kafka brokers.

Since multiple containers were required to be implemented, **docker-compose** tool is selected as it is aptly suited for the purpose of defining and sharing such multi-container applications.

- **Kafkajs** (“Apache Kafka client for Nodejs”)
- **Producer Script** Kafka producer is implemented to make asynchronous promise calls to stock-micro microservice

Monitoring Docker Containers Docker containers were monitored using docker desktop, using the log option under any specific container. For usage monitoring stats, stats option was used to view the statistics.

This could also be achieved using:

- **docker stats**: this command is used to view statistics such as memory usage, network usage for all the containers.
- **docker ps**: this command os used to list the containers to view statistics for those containers.
- **docker ps -all**: this command is used to list for all the containers.
- **docker container ps -filter “status=exited”**: this command is used to check the containers that have exited or stopped

- **docker container prune**: this command is used to remove container that exited, though they could be removed individually or at once using the bin symbol besides the container listing inside docker desktop, to bin symbol on the top right of all containers listing.
- **docker logs –since=30m 29ef5d137b2b**: this is a sample command that was used to get logs since past 30 min of container **29ef5d137b2b**, which is a kafka container id.
- Alternatively **docker logs –since=30m kafka6** could be used using the container name directly to retrieve the log where the **kafka6** is the container name.

Environment details for any specific container was used to view the setup details such server port number for a container, using the **inspect** option.

Observation Earlier KSQLDB was adopted to run queries on kafka stream, however as per application requirement, it was only needed to store user’s relevant order transactions. Implementation details, a container for building and deploying ksqldb server for real time stream processing of intended kafka topic(s).

5.3 *The Key Microservices Used in the Application*

Below are the details of the microservers used in the application and their invocation API.

- **Quote Service**: gets stock quote for a given day.
- **Profile Message**: gives an overview of company name, industry.
- **Candles Service**: provides high, low, open, close prices for a stock based on the interval earlier selected by user.
- **News Service**: provides top company news, source and URL to give insights into current market sentiments associated with company. In the application implementation top 10 news are displayed.
- **Income Service**: provides company liquidity over last years in terms of one of current ratios metrics, P/E: profit to earning ratios over last years. It could serve as important criteria for a user to find for analysing how risk or “future growth” associated when buying stock of that company. Low P/E ratio indicates high risk, can also be interpreted as low growth. Low P/E ratio indicates high risk, can also be interpreted as low growth.
- **RSI Service**: shows the relative stock index value for a stock based on the interval earlier selected by user up to the current time on any given day.
- **OBV Service**: shows the balance stock index value for a stock based on the interval earlier selected by user up to the current time on any given day.
- **STOCH Service**: shows the balance stock index value for a stock based on the interval earlier selected by user up to the current time on any given day.
- **ADX Service**: gives back average directional index value for a stock based on the interval earlier selected by user up to the current time on any given day.

- **CCI Service:** brings back on commodity channel index value for a stock based on the interval earlier selected by user up to the current time on any given day.
- **SMA Service:** gives back simple moving average value for a stock with interval of 5 min based the current time on any given day.
- **EMA Service:** presents the exponential moving average value for a stock with interval of 5 min based the current time on any given day.
- **TEMA Service:** provides the triple exponential moving average value for a stock with interval of 5 min based the current time on any given day.

Below is the name of the API (or function) as implemented in the application:

```
Quote Service: finnhub_quote_message ();
Profile Service: finnhub_profile_message ();
Income Service: finnhub_income_message ();
News Message: finnhub_news_message ();
Candles Service: finnhub_candles_message ();
RSI Service: finhubb_rel_index_message ();
OBV Service: finnhub_obv_message ();
STOCH Service: finnhub_stoch_message ();
ADX Service: finnhub_adx_message ();
CCI Service: finnhub_CCI_message ();
SMA Service: finnhub_SMA_message ();
EMA Service: finnhub_EMA_message ();
TEMA Service: finnhub_TEMA_message ();
```

5.4 The Prediction Services Used in the Application

Closing Price Prediction This is Closing Price Prediction for user's selected ticker was obtained by analysing the closing price data obtained from OHLC data using deep learning approach. This microservice was used to predict the subsequent 10 min closing prices. The Closing Price Prediction is implemented as a Nodejs module and the API is listed below:

```
get_ClosingPrice_Prediction
```

Observations The following are the main observations about the closing price prediction service.

- Two methods were implemented and compared for price prediction: Training accuracy and **LSTM timestep neural network** approach lead to better results as compared to **Recurrent neural network**;
- **minmaxscaler** module was as feature scaling method in the data preprocessing step to normalize to ensure that the neural network model's prediction the different in magnitude of closing price values

News Sentiment Prediction Gives the users valuable insights on top news obtained based of ticker user selected on the client side form, sentiment analysis was performed on news summary data obtained from all top news category To achieve this it is required to have **Tokenization function** which is performed as data preprocessing steps before applying the analyzer on each news summary.

Observations The following are the main observations about the news sentiment prediction service.

1. Three stemming were implemented and compared such as porter **porterStemmer**, **Lancaster Stemmer**, **Snowball Stemmer** were implemented and compared for price prediction:
2. Training accuracy and **LSTM timestep neural network** approach lead to better results as compared to Recurrent neural network;

5.5 *The Client Side of the Application*

Next, we discuss how the client side of the application is implemented.

1. On the client side stocks charts were rendered using to provide end users with instant insight into variation in stock prices, based on ticker and interval criteria set by a currently logged in user.
2. Stock fundamental data for basic company overview we stock quote, p/e ratios, top 10 news and associated market sentiment classified as bearish, bullish, neutral. Bearish category indicated pessimistic sentiment associated with a stock news summary indicating a stock price would fall in future, where a or bullish category indicated optimistic sentiment indicating a stock price would fall in future.
3. User could hover over any chart on the dashboard page and click on data points on any chart to initiate a buy or sell transaction. In this context, user clicks on a canvas element on HTML5 rendered using express handlebars. where charts are rendered using javascript functions. Threshold values were indicated for buy and sell, or to indicate a trend on in the technical indicator charts (OBV, ADX, SMA, EMA, TEMA) for providing baseline functionality support in user's decision making while placing order.
4. Due to asynchronous nature of promises, kafkajs library supports for consumer configuration on server side, kafka message topics could be sent using websocket only using a interval based approach, where 1 min is least interval limit. However on client side it appeared that end users would need to wait. Hence producer script was designed to sending stock data as kafka topics over the desired interval (1, 5, 10 min). On the other had consumer was designed to send the consumed message more frequently, sending the same consumed message obtained for say 1 min every 20 s, until producer would have write to the topics in next 1 min. We show a sample of this script in Fig. 3.


```

const {LineCh} = require("./create_charts");

const kafka = new Kafka({
  clientId: "kafka_consumer",
  groupId: "kafka_consumer_group",
  brokers: ["localhost:9092", "localhost:9093", "localhost:9094"],
});

var ticker;
const consumer = kafka.consumer({
  groupId: "kafka_consumer_group",
  allowAutoTopicCreation: false,
});
const topics = [
  "alpha_intraday_topic"
]; /*["alpha_intraday_topic", "alpha_profile_topic", "alpha_quote_topic", "alpha_income_topic",
"finnhub_profile_topic", "finnhub_quote_topic", "finnhub_candles_topic", "finnhub_income_topic", "alpha_rsi_topic",
"alpha_stoch_topic", "alpha_adx_topic", "alpha_obv_topic", "finnhub_rsi_topic", "finnhub_stoch_topic", "finnhub_adx_topic", "finnhub_
"alpha_sma_topic", "alpha_ema_topic", "alpha_tema_topic", "finnhub_sma_topic", "finnhub_ema_topic", "finnhub_tema_topic"]
*/

```

Fig. 3 Kafka producers and consumer

5. To keep client side light weight, most of the stock data processing was done on the nodejs microservices modules.

5.6 *Firestore Cloud Database*

1. **User Authentication:** As security remains to be one of majors concern for cloud service provider, in the proposed web application firebase backend authentication is used. Firebase backend authentication services were used to create user account, verify sign in or sign out. At any given point in the web application only user's display name was shared between the front end HTML pages. The user has to first login to the platform using the assigned credentials as shown in Fig. 4.
2. **textbfOrder Processing:** When User would click on a data point on any of the charts, it would lead to subsequent order page requesting user to submit, and then confirm the details of order. Two types of transactions are allowed either Buy or Sell.

In both the options user were only required to mention the quantity, and select suitable option from buy or sell from drop down in the UI as the price for data point is auto-filled in the form. Upon submission it would lead to a conformation page showing the total order value, current portfolio balance. The ticker selection choice is shown in Fig. 5 and the user ticker fundamental page is shown in Fig. 6.

5.7 *Challenges*

The challenges faced during the implementation of the trading application are:

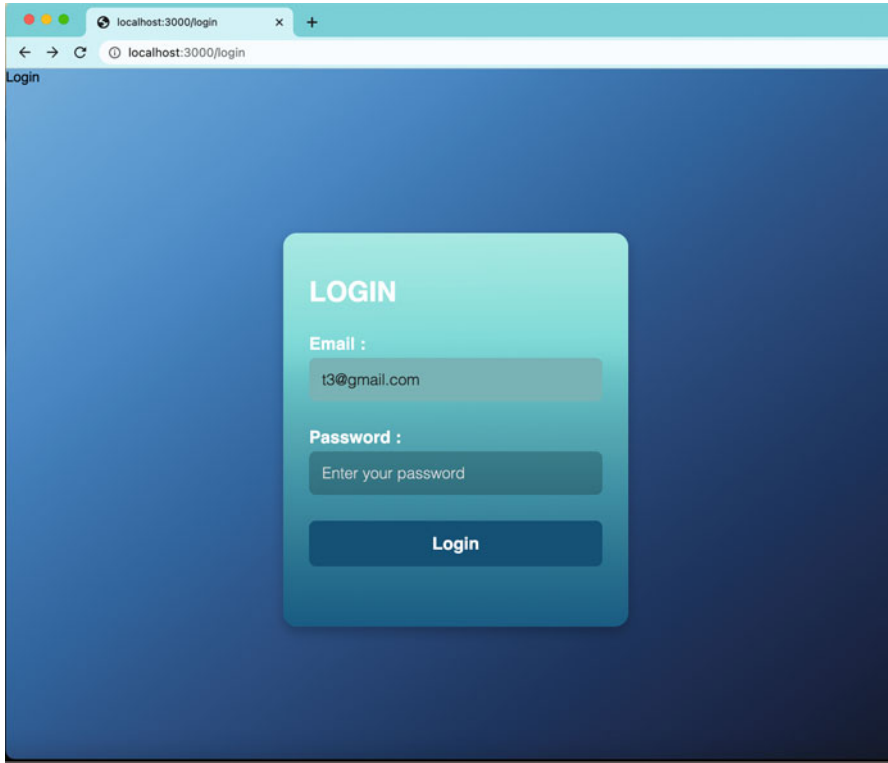


Fig. 4 Web application UI login webpage

1. It was observed that due to the unavailability of Kafka broker, it was required to redeploy the kafka, zookeeper containers. Hence, the number of Kafka brokers, also referred as storage layer, was increased to 3 brokers. Usually at least one of the brokers is available at any given time, sometimes the leader election (managed by Kafka) takes more time.
2. Lack of proper documentation support, examples for some of nodejs libraries.

6 Conclusions and Future Work

A light weight, cost effective, secure and scalable web application was created as a cloud service, emphasising on cloud native infrastructure considerations such as on backend, cloud database, Firebase was used for scalability with no downtime. In the middleware, kafka broker containers were fully managed by docker, that were easily, quickly deployed on cloud. Kafka producer consumer were implemented for real-time streaming of stock data as Kafka topics. Kafka was preferred in the

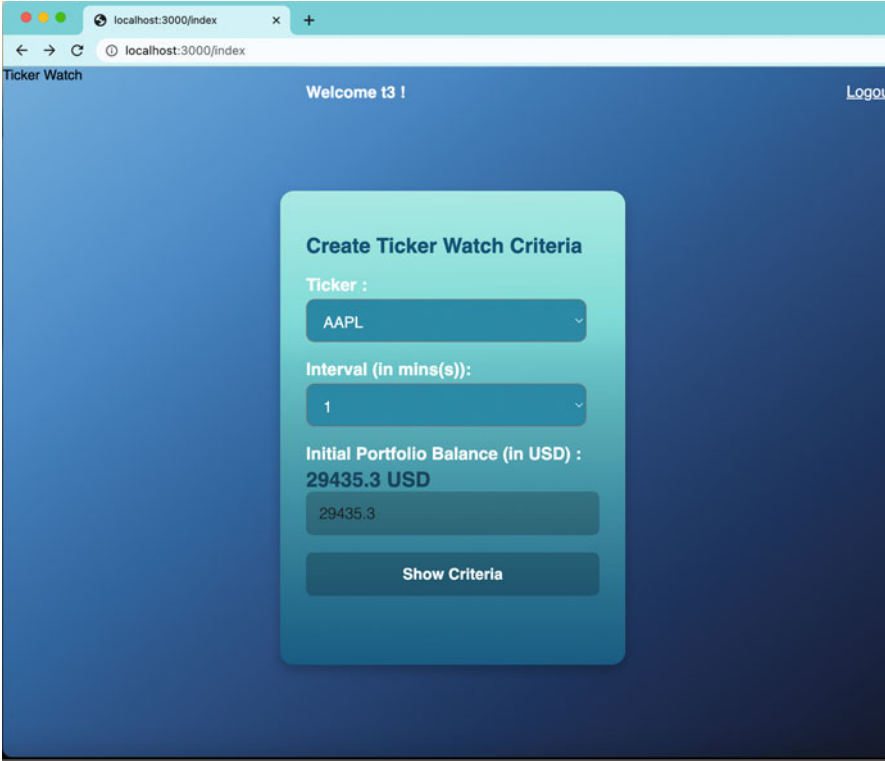


Fig. 5 User ticker selection page

middleware since load balancing, also known as scaling out, is managed internally by kafka based on partitioning and rebalancing. Websocket was used for faster client server communication to HTTP. In contrast to REST API calls, Microservices were created ensure scalability in terms of software development requirement. With low coupling between the microservices, a SaaS provider wishes to scale, can add more adding any modules in future depending on business requirements. For startup SaaS service providers planning to scale, dealing with high volume of sensitive data, these considerations would be helpful for providing high performance, secure and scalable framework, keeping end users experience at heart.

6.1 Future Directions

The proposed architectural framework could be applicable to other time-sensitive business verticals, such as e-commerce, and hospitals, amongst others, in which Kafka could be configured to fetch more data as per the data consumption require-

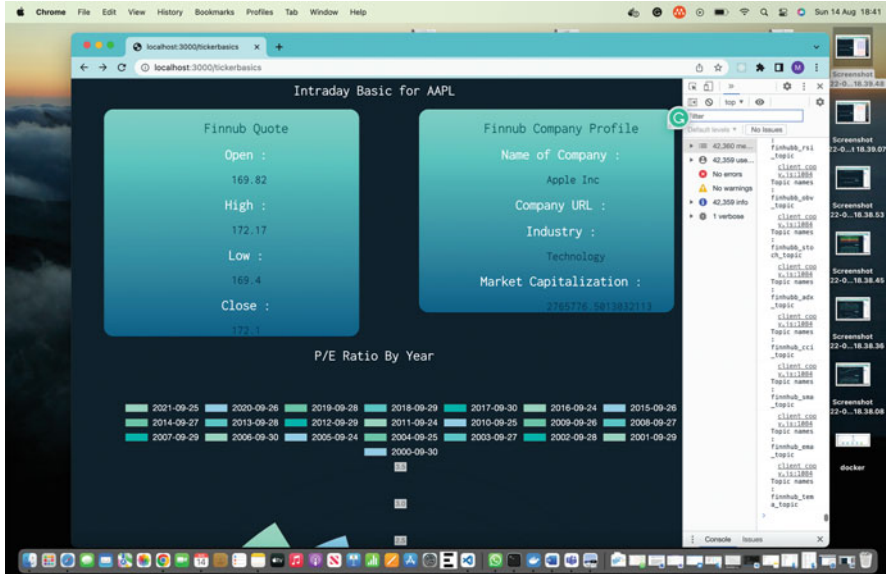


Fig. 6 User ticker fundamental page

ment. In future work, from an end-user perspective, integrations to platforms such as Slack could be built, so that users could get instant notifications on stock prices. From a SaaS provider's perspective, Big Query integration could be built to analyse user google analytics metrics when the user base expands to that of an enterprise to understand user behaviour [16]. On the SaaS provider end, Jira integration could also be built for easier tracking of any user management issues (in the current context of the project). Additional load balancing measures could be taken using the NGINX load balancer or Kubernetes load balancer to manage containers effectively from Kafka's internal load balancing [17]. To enhance the analytics of the application, distributed machine learning could be utilized with acceleration techniques to speed the processing [18, 19]. Finally, to speed up the communication between Kafka and Microservices, techniques for improving the TCP performance can be applied [20].

Acknowledgments This work is partially funded by Chinese Academy of Sciences President's International Fellowship Initiative (Grant No. 2023VTC0006), National Natural Science Foundation of China (No. 62102408), Shenzhen Science and Technology Program (Grant No. RCBS20210609104609044), and Shenzhen Industrial Application Projects of undertaking the National key R & D Program of China (No. CJGJZD20210408091600002). We also declare that this work has been submitted as an MSc project dissertation in partial fulfilment of the requirements for the award of degree of Master of Science submitted in School of Electronic Engineering and Computer Science of Queen Mary University of London, UK is an authentic record of research work carried out by Mousumi Hota (first author) under the supervision of Sukhpal Singh Gill (last

author) and refers other researcher's work which are duly listed in the reference section. This MSc project dissertation has been checked using Turnitin at Queen Mary University of London, UK and submitted dissertation has been stored in repository for university record.

References

1. Chung, J., Choe, H., & Kho, B.-C. (2008). The impact of day-trading on volatility and liquidity. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.1855759>
2. Stopford, B., & Newman, S. (2018). Concepts and patterns for streaming services with Apache Kafka designing event-driven systems. [online] Available at: [https://sd.blackball.lv/library/Designing_Event-Driven_Systems_\(2018\).pdf](https://sd.blackball.lv/library/Designing_Event-Driven_Systems_(2018).pdf)
3. Salah, T., Jamal Zemerly, M., Yeun, C. Y., Al-Qutayri, M., & Al-Hammadi, Y. (2016). The evolution of distributed systems towards microservices architecture. In *11th international conference for internet technology and secured transactions (ICITST)* (pp. 318–325).
4. Shah, S. D. A., Gregory, M. A., & Li, S. (2021). Cloud-native network slicing using software defined networking based multi-access edge computing: A survey. *IEEE Access*, *9*, 10903–10924.
5. Ammi, M., Adedugbe, O., Alharby, F. M., & Benkhelifa, E. (2022). Leveraging a cloud-native architecture to enable semantic interconnectedness of data for cyber threat intelligence. *Cluster Computing*, *25*, 3629–3640.
6. Valdez, M. G., & Guervós, J. J. M. (2021). A container-based cloud-native architecture for the reproducible execution of multi-population optimization algorithms. *Future Generation Computer Systems*, *116*, 234–252.
7. Saxena, H., & Pound, J. (2020). A cloud-native architecture for replicated data services. In *12th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 20)*.
8. Fernando, J. (2019). Relative Strength Index – RSI. [online] Investopedia. Available at: <https://www.investopedia.com/terms/r/rsi.asp>
9. Hayes, A. (n.d.). Stochastic oscillator. [online] Investopedia. Available at: <https://www.investopedia.com/terms/s/stochasticoscillator.asp>
10. Hayes, A. (n.d.). On-Balance Volume (OBV). [online] Investopedia. Available at: <https://www.investopedia.com/terms/o/onbalancevolume.asp>
11. Investopedia. (n.d.). Exponential Moving Average - EMA. [online] Available at: <https://www.investopedia.com/terms/e/ema.asp>
12. Mitchell, C. (n.d.). Triple Exponential Moving Average (TEMA) definition. [online] Investopedia. Available at: <https://www.investopedia.com/terms/t/triple-exponential-moving-average.asp>
13. Xu, M., Song, C., Hager, S., Gill, S. S., Zhao, J., Ye, K., & Xu, C. (2022). CoScal: Multi-faceted scaling of microservices with reinforcement learning. *IEEE Transactions on Network and Service Management*, *19*, 3995–4009.
14. Balalaie, A., Heydamoori, A., & Jamshidi, P. (2016). Migrating to cloud-native architectures using microservices: An experience report. In A. Celesti & P. Leitner (Eds.), *Advances in service-oriented and cloud computing. ESOC 2015*. Communications in computer and information science (Vol. 567). Cham: Springer. https://doi.org/10.1007/978-3-319-33313-7_15
15. Long-Polling vs WebSockets vs Server-Sent Events, Online Link <https://systemdesignbasic.wordpress.com/2020/02/01/12-long-polling-vs-websockets-vs-server-sent-events/>
16. Iftikhar, S., Gill, S. S., Song, C., Xu, M., Aslanpour, M. S., Toosi, A. N., et al. (2022). AI-based fog and edge computing: A systematic review, taxonomy and future directions. *Internet of Things*, *21*, 100674.
17. Gill, S. S., Xu, M., Ottaviani, C., Patros, P., Bahsoon, R., Shaghghi, A., et al. (2022). AI for next generation computing: Emerging trends and future directions. *Internet of Things*, *19*, 100514.

18. Gajjala, R. R., Banchhor, S., Abdelmoniem, A. M., Dutta, A., Canini, M., & Kalnis, P. (2020). Huffman coding based encoding techniques for fast distributed deep learning. In *Proceedings of the 1st Workshop on Distributed Machine Learning*.
19. Abdelmoniem, A. M., & Canini, M. (2021). DC2: Delay-aware compression control for distributed machine learning. In *IEEE Conference on Computer Communications (INFOCOM)*.
20. Abdelmoniem, A. M., & Bensaou, B. (2017). Enforcing transport-agnostic congestion control in SDN-based data centers. In *IEEE 42nd Conference on Local Computer Networks (LCN)*.
21. Kuepper, J. (n.d.). An introduction to day trading. [online] Investopedia. Available at: <https://www.investopedia.com/articles/trading/05/011705.asp>
22. Mitchell, C. (n.d.). Commodity Channel Index - CCI definition and uses. [online] Investopedia. Available at: <https://www.investopedia.com/terms/c/commoditychannelindex.asp>
23. Hayes, A. (n.d.). Simple Moving Average - SMA. [online] Investopedia. Available at: <https://www.investopedia.com/terms/s/sma.asp>
24. Lee, C., & Paik, I. (2017). Stock market analysis from twitter and news based on streaming big data infrastructure. In *IEEE 8th international conference on awareness science and technology* (pp. 312–317).

Part II
Architecture, Systems and Services

Efficient Resource Allocation in Virtualized Cloud Platforms Using Encapsulated Virtualization Based Ant Colony Optimization (EVACO)



Nirmalya Mukhopadhyay , Babul P. Tewari , Dilip Kumar Choubey ,
and Avijit Bhowmick

Abstract Virtualization in cloud computing ensures maximum utilization of the computational resources by creating virtual instances. In this paper, we have jointly considered virtualization and ant colony optimization (ACO) to propose a novel encapsulated virtualization-based ACO (EVACO) technique that allocates the computational resources efficiently. We have developed an efficient mathematical model that modifies the traditional ACO and encapsulated it in virtualized cloud platform. The objective is to optimize the resource allocation so that execution can be efficient. The novelty of the proposed model has been established through extensive simulations and comparative study. Our algorithm took less execution time in compare to some popular benchmark algorithms. Moreover, the number of iterations and virtual resources required to complete tasks were less than that of other algorithms. It establishes the optimality of our proposed approach.

Keywords EVACO · Virtualization · Cloud computing · Ant colony optimization · Execution model in cloud · Cloud cost model

1 Introduction

Cloud computing has been a twenty-first century marvel that has occupied the entire information technology (IT) world and also many of the non-IT enterprises has shifted their business information to cloud. It has appeared as a virtue to both

N. Mukhopadhyay · B. P. Tewari (✉) · D. K. Choubey
Department of Computer Science & Engineering, Indian Institute of Information Technology,
Bhagalpur, India
e-mail: nirmalya.cse.2103004@iiitbh.ac.in; bptewari.cse@iiitbh.ac.in;
dkchoubey.cse@iiitbh.ac.in

A. Bhowmick
Department of Computer Science & Engineering, Budge Budge Institute of Technology, Kolkata,
India

enterprises and academic users [1]. The reasons being obvious that it has helped the business fraternity to reduce their IT overhead. The introduction of this new age technology has revolutionized the concept of computational resource utilization. Hence, a major change in the computational approach has been observed. Different computing devices such as processing units, memory units, storage components, networking segments are virtualized and these virtual instances are used to create cloud infrastructural environment [2]. It has been noticed that the total cost of ownership (TCO) has been in the affordable range for the small and medium size enterprises [3].

Resource optimization in cloud computing has been incorporated in many different ways. Many algorithms have been built and many computational and mathematical models have been designed such as particle swarm optimization, genetic algorithm, Nash equilibrium, cooperative game-based optimization etc. [4–7]. In this context, ant colony optimization (ACO) is also a popular technique for achieving the optimization solution in an efficient manner. It follows probabilistic slant to discover the finest set of solutions from a puddle of solutions. Researchers have developed computational logic to create artificial ants through programming. These artificially programmed agents (resemblance with the behavior and properties of biological social insect ant) ensure a better solution after every iteration through predominant paradigm [8, 9]. Finally, the best solution is obtained and applied for enhancing the overall performance of the system. In our proposed approach, we have investigated resource optimization through a modified ACO technique to optimize the execution cost and time through an efficient resource selection mechanism.

While virtualization will help in mimicking the computational resources into virtual instances [4, 10], and logically isolating and segregating them for keeping individual existence for participating into optimization of resources for cloud computing; the ACO algorithm will be in action to find out the best probable solutions for allocating the resources to the virtual instances. Virtualization will help in maximizing the utilization of the available resources [1, 11] whereas, ACO will allocate the required resources to incoming tasks. So, in this study, we have encapsulated the virtualization technology and ACO algorithm with the objective to reach to an optimized virtual resource allocation method that will efficiently compute all the tasks without affecting the system performance.

Cloud computing is used to store secure end-to-end encrypted IoT data on multiple servers and can be accessed online. To avoid the dependency of internet based centralized cloud storage and edge devices, fog computing creates local networks for decentralized IoT ecosystem to decide whether to process the data locally or remotely. This means, fog computing enhances the opportunity to analyze and process offline data if access to the cloud is not stable or possible. Hence, efficient resource allocation in virtualized cloud platform unfolds a gateway for fog computing to properly place fog nodes into edge platforms that can provide unprecedented processing speed on sensitive and real time operations for data analytics in IoT.

In this study, we have addressed a major issue: whether the resource allocation policy can be optimized in virtual cloud platform or not. So, our objective is to

find out the most feasible resource allocation policy among a pool of solutions that provides best cloud execution model to the cloud users. To ensure this selection, encapsulated virtualization-based ant colony optimization (EVACO) algorithm is proposed for partially virtualized cloud platform. This algorithm ensures the selection of the improved solution in such a way, so that least number of iterations are required to complete the execution. So, in our proposed approach, first the pool of virtualized resources is created as an initial population and then an independent resource will be selected through EVACO algorithm to complete the assigned task with minimum amount of time without affecting the system performance.

The rest of the paper is organized as follows: Section 2, describes the existing works. We have explained the traditional methods in Sect. 3. Our proposed mathematical model and associated EVACO algorithm is formulated in Sect. 4, which is followed by the result analysis in Sect. 5. Finally, we conclude our discussion in Sect. 6.

2 Motivation

Computing has changed its paradigm from time to time. The main objective of this transformation is to efficiently solve real world problems. Furthermore, computing technologies became a part of daily activities. In this regard, computation needs to be intelligent and easily accessible to the users. The variants of distributed system focus on reliable, available and demand-based execution of tasks. Allocation of resource towards effective computation has always been an area of research and development. Different algorithms for different environments and dependencies have been proposed and implemented to enhance the execution process. Virtual cloud platforms make online analysis of IoT data and fog nodes do it offline. In both the platforms, resources are allocated to resolve tasks in competent manner. We started our research with the motivation to make resource allocation optimal for virtualized cloud platforms, so that we get most feasible solutions with unparalleled speed and performance.

3 Related Study

There have been considerable research works focusing on optimizing and analyzing the resource allocation of cloud computing. In such context [12], suggested a task scheduling algorithm based on the ant colony optimization algorithm (ACO) that efficiently allocates cloud resources to users' jobs within the virtual machines using diversity and reinforcement techniques [13] found that the typical cloud-based strategy was proven to be unable to meet the requirement of low-latency execution of multiple devices in an edge area. Therefore, they planned to incorporate ACOA-based internet of things (IoT) approach that can lead to the number of edge

devices to create an edge data centre. Finally, they proposed a two-level scheduling optimization scheme that will produce an efficient output in timely manner [5] defined a large-scale peer to peer (P2P) grid system that implements an ACOA to identify the location of required resources. The authors of [14] have proposed an efficient IaC-based resource allocation framework for simulated virtual cloud platforms that provides high throughput and effortless cloud resource configuration management.

Mishra et al. [15] demonstrated the re-engineering approaches to migrate the ACOA to modern Intel multi-core architectures to mitigate the factors that has an impact on hardware performance. On similar research [13], presented a new open computing language (OpenCL) based ACOA and [16] have discussed novel parallel algorithms of the well-known ACOA on the multi-core platforms of Intel Xeon Phi co-processor.

The authors of [8] established a strategy for efficient software module clustering in which dependent modules are placed together inside a cluster. According to [2], the computational complexity of ACO if compared to that of genetic algorithm (GA), crops better result as ACO can optimize the services to a higher degree [9] projected an adaptive learning model based on ACO Algorithm to improve stint, price and eminence factors to realize the optimal resource allocation.

On contrary, our research focuses on optimizing the resource allocation towards completion of cloud tasks in a partially virtualized cloud platform efficiently without hampering the SLA and system performance through encapsulating the ACO algorithm with virtualization technology. In this context, we propose a novel EVACO algorithm along with supportive mathematical cloud execution model. Our research shows better performance than that of existing algorithms, which have been proved through extensive simulations.

The following table displays the comprehensive comparison between different researches discussed above (Table 1).

4 Traditional ACO

In order to formulate the proposed EVACO, we have analyzed the basic approach of ACO. There has been a considerable study to incorporate virtualization techniques to increase the infliction of computational resources. Conversely, ACO has also proven to be a great solution towards resource allocation methods.

In ACO algorithm, each ant obtains a starting node that can be considered as it's nest [5, 17, 18]. From this node, the ant selects the next node based on the rules of the algorithm. The ant completes its journey by traversing all the nodes only once and finally returns home at last. The ants can travel the nodes either concurrently or successively. During the journey, each ant places a convinced volume of pheromones on the route. The amount of pheromone to be poured down in the routes depends on the quality of the path selected by the ant; a petite path usually fall-outs with

Table 1 Comprehensive comparison of related studies

Paper	Problem found	Proposed solution	Remarks
[12]	Inefficient resource allocation in cloud platforms	Diversity and reinforcement-based scheduling techniques optimized through ACOA	Provided better result than existing algorithms in conditional cases
[8]	Lack of dependency in cluster-based cloud platforms	Efficient software module clustering strategy	Resource allocation and response time improved
[15]	Reduced hardware efficiency & performance	Migrating ACO to processor-based operations.	Performance of hardware increased
[13]	High latency in execution through edge devices	Creation of edge data center using OpenCL	Latency dropped significantly
[16]	Reduced hardware efficiency & performance	Developed ACOA for multicore Intel Xeon Phi coprocessor	Performance of hardware increased
[5]	Location transparency of cloud resources	Developed large-scale P2P grid system embedded with ACOA for resource location finding	Identification of location becomes efficient
[9]	High price of computing	Proposed adaptive learning model based on ACOA	Improved stint and price
[14]	Reduced performance and throughput because of configuration management overhead during cloud resource provisioning	Proposed an efficient IaC-based resource allocation framework	Improved throughput and performance because of programmed provisioning

more pheromone. The precipitated pheromone undergoes disappearance which is best known as evaporation.

The probability of selecting the succeeding node j by the Ant to arrive at from the current node i is computed as follows [15, 16, 18]:

$$p(c_{ij}|s^p) = \frac{\tau_{ij}^{\alpha*} \eta_{ij}^{\beta}}{\sum_{c_{ij} \in N(s^p)} \tau_{ij}^{\alpha*} \eta_{ij}^{\beta}} \quad (1)$$

where, s^p = a partial solution, N is the list of all existing routes from node i to every neighbor node which are yet to be traversed by the ant, c_{ij} is the route from i to j , p is the probability of incidence, t_{ij} is the quantity of pheromone follow on c_{ij} , h_{ij} is calculated as some empirical factor, $\eta_{ij} = Q/d_{ij}$, where d_{ij} is a remoteness factor amid nodes i and j , Q is a constant weight and α and β are the algorithmic constraints.

4.1 Apprising Pheromone Value

The Ants will update the value of the pheromone on the routes connecting the nodes according to the following formula [2, 11, 16, 17]:

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (2)$$

where, m is the count of ants, $\Delta\tau^k = Q/L_k$ if the ant k traveled the path C_{ij}^k between nodes i and j ; Q is a persistent weight (it is the set of all neighbor nodes), and L_k is the distance covered by the ant k during travel and is represented as $\Delta\tau^k = 0$.

4.2 Evaporation

After finishing the n th trip by the Ants, evaporation will take place in all the available routes between the nodes. This is given by the formula as follows:

$$\tau_{ij}^n \leftarrow (1 - \xi) * \tau_{ij}^n \quad (3)$$

where, $\xi \subseteq (0,1]$ is the evaporation factor.

5 Proposed Solution

We propose for a modified version of the basic ACO algorithm by encapsulating it in virtualized platform. We call it encapsulating virtualization-based ant colony optimization (EVACO). To implement our algorithm, first, the available computing resources need to be virtualized to create the initial population. These virtualized resources provide maximum resource utilization percentage through isolated instances. Then, we will apply ACO algorithm on the entire pool of isolated yet correlated and fully functional virtual resources. This, in turn will provide us the optimum solution in minimum number of iterations. The entire process will not hamper the overall performance of the cloud platform. To implement our algorithm, we have taken a step-by-step approach. We have described each step in the following discussion:

5.1 Formulating Primary Population for Virtualized Computing Resources

A control variable in the optimization model is the path taken by an ant in any stratum of an N-dimensional space. The N-dimensional array indicates the path taken by the ant. The array can be represented as:

$$VRP = (vr_1, vr_2, vr_3, \dots, vr_i, \dots, vr_n) \tag{4}$$

where, virtual resource pool (VRP) is a set of solutions, n is the total number of control variables, and vr_i is the i th control variable in a single VRP. As shown below, each control variable i selects a weight from a discrete domain V_i 's predetermined set of weights:

$$V_i = (v_{i,1}, v_{i,2}, \dots, v_{i,d}, \dots, v_{i,D_i}) \tag{5}$$

in which, $i = 1,2,3,\dots,N$; V_i is the encoded set of weights for i th control variable, $v_{i,D_i} = d$ th conceivable weight for the i th control variable, and $D_i =$ total number of conceivable weights for the i th control variable. The EVACO algorithm begins with generating a $M \times N$ matrix at random, where M is the population size of results and N represent the number of control variables. As a result, the pseudo-random solution matrix looks like this:

$$P_{VR} = \begin{bmatrix} VRP_1 \\ VRP_2 \\ \vdots \\ VRP_j \\ \vdots \\ VRP_M \end{bmatrix} = \begin{bmatrix} vr_{1,1} & vr_{1,2} & \dots & vr_{1,i} & \dots & vr_{1,N} \\ vr_{2,1} & vr_{2,2} & \dots & vr_{2,i} & \dots & vr_{2,N} \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ vr_{j,1} & vr_{j,2} & \dots & vr_{j,i} & \dots & vr_{j,N} \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ vr_{M,1} & vr_{M,2} & \dots & vr_{M,i} & \dots & vr_{M,N} \end{bmatrix} \tag{6}$$

in which P_{VR} is the population of virtual resources; VRP_j is the j th solution, $vr_{j,i}$ is the i th control variable of the j th solution, and M is the population size (i.e., the quantity of existing solutions to solve the task). The weight of $vr_{j,i}$ is arbitrarily nominated from a set V_i .

5.2 Apprising Pheromone to the Control Domain

The EVACO looks for information in various parts of the control space and adds it there. New solutions are created erratically depending on the statistical data available in the control domain. The EVACO algorithm assigns pheromone intensity value to each control variable's weight depending on the solution's fitness score.

The higher the pheromone quantity, the more suitable a solution is, and the other way around. In virtualization scenario, if a virtual resource is available and idle, then the best solution is to allocate that resource to solve a problem to increase the utilization of resources in an optimized way. So, the idle index (ϕ) will be used in exchange of pheromone during the solution. The higher idle index is for a virtual resource, the higher the chances of allocating that resource to a pending task.

To apportion pheromone to the discrete domain, N arrays of size $1 \times D_i$ are used, each array is allotted to one control variable as follows:

$$P_i = (p_{i,1}, p_{i,2}, \dots, p_{i,d}, \dots, p_{i,D}) \quad (7)$$

where, P_i = the pheromone matrix for the i th control variable and $p_{i,d}$ = the pheromone intensity of the d th probable weight of the i th control variable. At the start of the algorithmic refinement, the elements of the matrix P_i equate zero. In virtualization platform, as the idle index is changed for a virtual resource, it becomes either not available or any other higher idle index resource is chosen for next pending task. So, the availability index (σ) is equivalent with evaporation coefficient (ϵ).

In basic ACO algorithm, pheromone intensity for the d th possible weight of the i th control variable is updated as follows [3, 15]:

$$p_{i,d}^n = (1 - \epsilon) * p_{i,d} + \sum_{j=1}^M \Delta c_{i,d}^j \quad (8)$$

in which, P^n = the updated intensity of pheromone of the d th possible weight of the control variable, ϵ = the rate of evaporation and $\sum_{j=1}^M \Delta c_{i,d}^j$ = the quantity of pheromone laid on the d th possible weight of the i th control variable by the j th ant.

The weight of $\sum_{j=1}^M \Delta c_{i,d}^j$ resembles to the fitness score of the j th solution, and is assessed for our proposed EVACO algorithm as follows in a minimization problem:

$$\Delta c_{i,d}^j = \begin{cases} \frac{Q}{F(VRP_j)} & \text{if } vr_{j,i} = v_{i,d} \\ 0 & \text{if otherwise} \end{cases} \quad (9)$$

where $j = 1, 2, \dots, M$; $i = 1, 2, \dots, N$; $d = 1, 2, \dots, D_i$; Q is a constant weight and $F(VRP_j)$ is the fitness weight of the j th solution.

In the EVACO algorithm, we can replace the pheromone weight $p_{i,d}$ with idle index (ϕ) and the evaporation coefficient (ϵ) with availability index (σ). So, the modified equation can be derived from Eqs. (8) and (9) as follows:

$$\text{maximize } (\phi) = \left[(1 - \sigma) * \phi + \sum_{j=1}^M \frac{Q}{F(VRP_j)} \right] \quad (10)$$

This equation will consider a lower edge weight which is bare minimum for idle index as we have used a floor function to set the lower limit. Any weight below a predefined constant will be disregard and hence the performance degradation cannot occur. If and only if the idle index of a computational resource is higher than the lower bound, then EVACO will consider that resource for further update.

5.3 Generation of Updated Solution

There is always an infinitesimal probability of never nourishing the condition $p(c_{ij} | s^p) \geq n_j$, which can be deduced from equation number 1. So, in this research, we have used some amendment in the traditional ACO Algorithm. For the given node i , we create a random number $n_j \in [0,1)$ and then compare that to the moving sum. The result is updated whenever is recalculated. When $n_j p(c_{ij} | s^p)$ satisfies, considered as the index of the next node j to travel. Essentially, this difference is only noticeable in the beginning phases of the test, when the amount of pheromone trailing on the various pathways is fairly similar. The summation of the probabilities of the conceivable weights of each control variable is equal to one. So, we write:

$$\sum_{d=1}^{D_i} \Psi_{i,d} = 1 \quad (11)$$

The weights of the control variables of an updated result are arbitrarily elected depending on the assessed likelihood. To do so, we calculated a cumulative probability for each potential weight of each judgement variable, as shown below:

$$\Upsilon_{i,r} = \sum_{d=1}^r \Psi_{i,d} \quad (12)$$

in which, $\Upsilon_{i,r}$ is the accretive likelihood of the r th possible weight of the i th control variable. Then, an arbitrary weight (ω) is inferred within the range of $[0,1]$. Depending on the comparative values obtained, weights are selected. For example, if the weight of ω is less than the weight of $\Upsilon_{i,r}$, $\Upsilon_{i,1}$ is selected; otherwise, the r th weight is selected in a pattern where $\Upsilon_{i,r} - 1 < \omega < \Upsilon_{i,r}$.

5.4 Mutating EVACO to Optimize Further

The worst possible situation that may occur in any optimization algorithm is when it gets trapped in some local optimal loop. Hence, the notion of mutation is brought from the genetic algorithm (GA) to avoid being locked into the local minima. When an ant has finished its trip, it will begin the mutation process based on the mutation probability. A node is randomly eliminated from the tour and is replaced by another

Table 2 Symbols used in EVACO algorithm and their descriptions

Symbols	Description
M	Initial population size (set of all virtual instances)
P_{VR}	Randomly generated population of virtual instances
n	# control variables
V_i	# possible weights for control variable i
TQ	Task queue (used as source node)
R_v	Virtual resource (used as destination node)
VRP	Decomposed tasks (used as ants)
AL	Availability list

randomly picked node from the same group. Finally, the randomly chosen node is added into the (m I) positions. The new solution of the afflicted ant is the shortest tour of all the feasible tours, including the initial route. As a result, the total number of iterations reduces up to a great extent resulting optimal solution. The outline of the proposed EVACO is represented in Algorithm 1. Note that, Table 2 represents different notations used in this algorithm.

Algorithm 1: Proposed EVACO

Input: $M, P_{VR}, N, V_i, TQ, R_v, VRP, AL$

Output: Optimal resource allocation to the decomposed task

Initialize $M=N=AL=0$

begin

generate M initial population P_{VR} of possible solutions randomly
while termination criteria stand false **do** {

evaluate fitness values for all the existing solutions

assign idle index to decision variables based on fitness values

finalize availability index to select if a resource will enter
 AL or not

allocate idle resources to VRP

update AL by removing the already allocated resources

for i : = 1 to N **do** {

for d : =1 to V_i **do** {

update idle index of possible value d_{ij} for decision variable i
 based on availability index and fitness value

evaluate selection probability of possible value d_{ij}

 }

}

for k : = 1 to M **do** {

for i : = 1 to N **do** {

select a random value r_{ki} for i^{th} decision variable among all
 possible values based on their probabilities

 }

}

}

select a solution if the fitness value is higher than initial
 solution

update P_{VR}

mutate to exit the local minima for optimize further

}

5.5 *Flowchart and Service Model*

A flowchart is a pictorial representation of flow of control. For our proposed EVACO algorithm, we have presented a flowchart that make the concept easier for implementation. Figure 1 below, is the technical flowchart for our algorithm.

We, moreover, have designed a service model for our proposal that shows how the subtasks are assigned with virtual resources efficiently using EVACO algorithm to produce optimal solution for the cloud customer. Figure 2 shows the service model for our proposal.

6 Result Analysis

In this section, we evaluate our proposed EVACO algorithm to prove its expediency. In order to evaluate the performance of the proposed approach, we have considered a methodological approach to create an environment for our simulations. We have developed a customized java program to implement the proposed EVACO algorithm. We have first discussed about the simulation set-up followed by the analysis of the results obtained from the simulations. Finally, the proposed EVACO algorithm is compared with some existing approaches from the literature.

6.1 *Simulation Set-Up*

Our experiment for evaluating the proposed EVACO algorithm and the proposed execution model requires both hardware and software set-ups. Using the hardware set-up, we have created a virtualized cloud platform. The detail of hardware set-up is listed in Table 3.

The software module for our simulation is installed using the hardware specifications mentioned in Table 4. As we have to apply our proposed algorithm in a virtualized platform, we have used VMWare workstation as type-II hypervisor. We have installed it on the top of Windows 10 operating system. After installing the workstation, we have created host virtual machines using ESXi server virtualization framework.

To complete the simulation, we have started with creating the virtual instances inside the physical computer system. We have used the set-up described in Table 3 for this step. We have implemented EVACO algorithm through a customized java code, running in our VMs. The primary intention is to find a set of optimum resources that can be allocated to each and every decomposed tasks. While implementing the code, we have taken care of all the constraints of EVACO algorithm. Our proposed algorithm is a direct enactment of our proposed mathematical model.

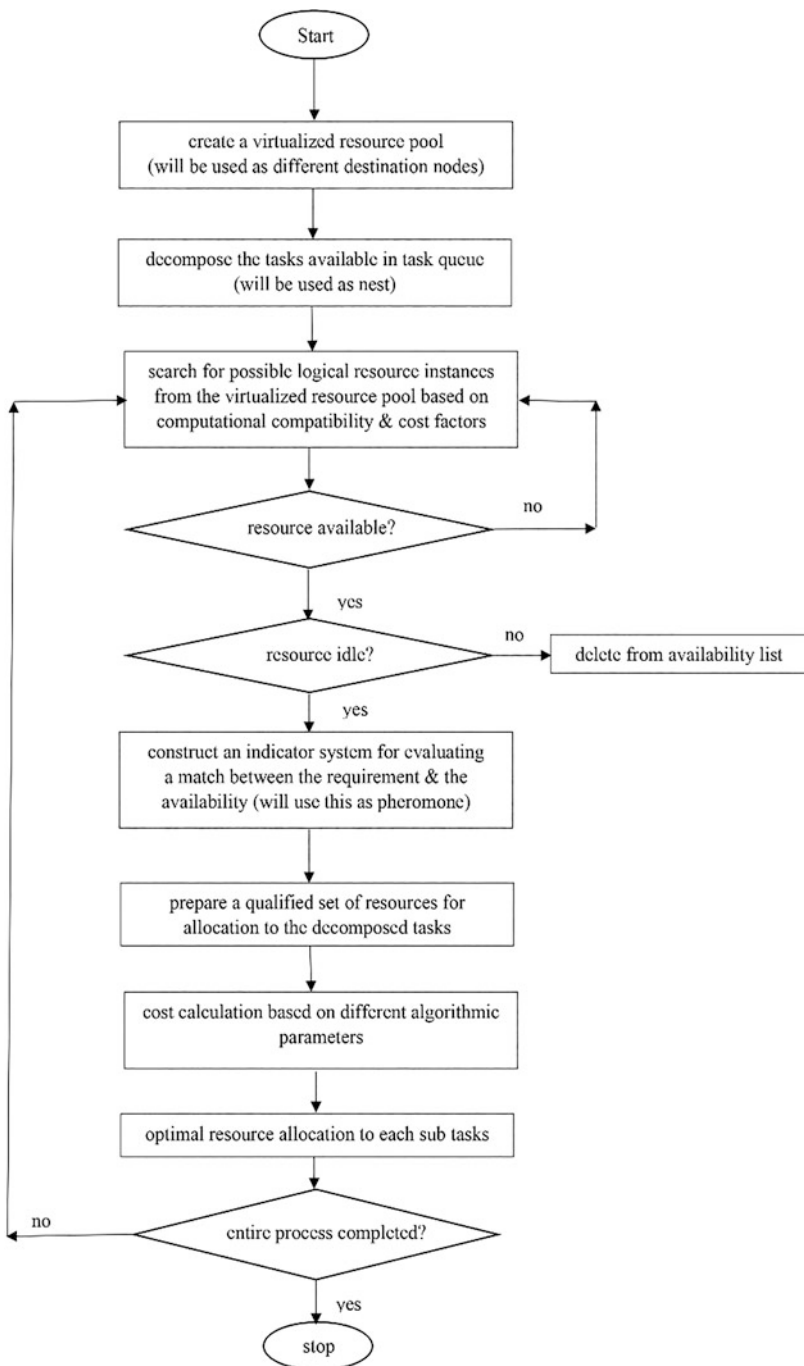


Fig. 1 Flowchart for EVACO algorithm

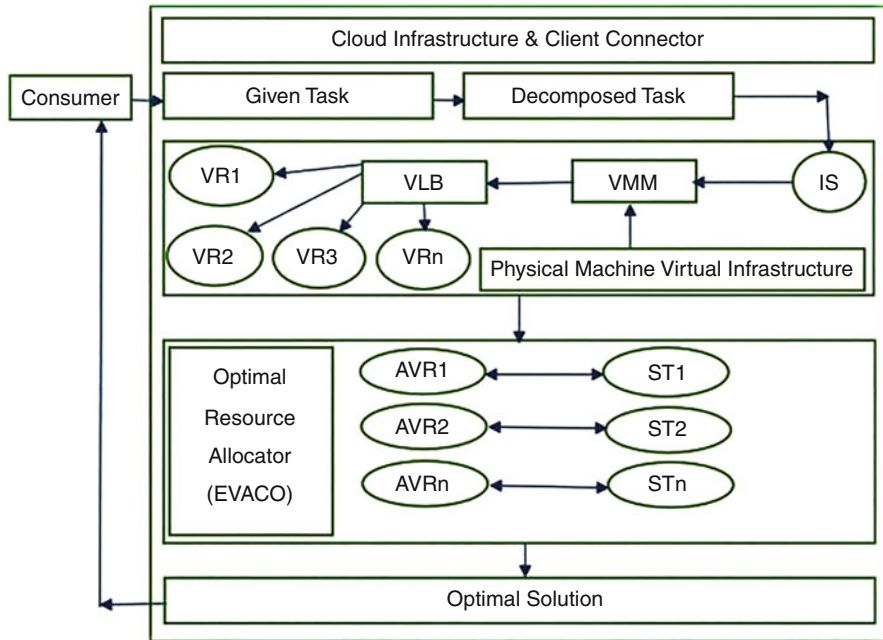


Fig. 2 Proposed service model

So, while writing the java code, we have taken care of all the required parameters for execution.

6.2 Simulation Result

Performance Evaluation of EVACO Algorithm

In order to evaluate the performance of our proposed EVACO algorithm, we focus on the parameter of resource allocation to a sub task and minimization of time to complete the entire process. The algorithm is designed to reach to the solution with minimum number of iterations. Additionally, through mutations it avoids of getting trapped inside the local minima.

We have started our experiment with 20 virtual resources and arbitrarily 186 number of subtasks. These virtual resources have been created inside a host. Our aim in the algorithm is that whenever a subtask will appear in the queue, it will be allocated to the resource that is most feasible. So, the virtual resources (also called nodes) are placed randomly in a predefined partially virtualized cloud platform.

The starting node, marked as 1 in Fig. 3a and Fig. 3b also chosen randomly and represents the comparative node traversal routes in two different iterations. From

Table 3 Details of hardware set-up for simulation

Parameters	Weight/range	Specification
# data centers	5	Simulated within the physical machine
# hosts	5	Running inside the host
RAM	16,384 MB	Double data rate V4
Host RAM	12,288 MB	Double data rate V4
GFX card	4096 MB	Provides additional processing power
Host N/W bandwidth	500–2000 MB/s	Network card of physical machine is used
# CPUs	[2, 8]	Intel i7 octa core 3.06 GHz
# VMs	40	Used VMWare workstation, ESXi v6.7
# vCPUs	[1, 4]	Assigned virtually to VM using VMM
Capacity of vCPU	[1000, 2500] MIPS	Assigned virtually to VM using VMM
Capacity of vRAM	[512, 4096] MB	Assigned virtually to VM using VMM
Capacity of VM bandwidth	[500, 2000] MB/s	Assigned virtually to VM using VMM
Direct attached storage	1,048,576 MB	SATA HDD
Virtual hard disk for host	358,400 MB	Virtual hard disk created & managed
VHD for VM	102,400 MB	Virtual hard disk assigned to VM
# cloud tasks	[100, 700]	Taken through ispell dataset ftp://gnu.mirror.iweb.com/
MIPS of vCPU	[100, 20,000]	vCPU speed measured in MIPS
Size of task files	[200, 400] MB	Located inside VHD
Size of output file	[20, 40] MB	Saved inside VHD

Table 4 Details of software set-up specifications

Parameters	Specification/description
File system	Standard virtual machine file system
Processor	Intel VT-x 64-bit x86 NX/XD bit enabled processor with at least 2 cores
Main memory	8 GB vRAM
Ethernet controller	1 GB
VHD	SCSI 350 GB
Boot partition	32 GB HDD with 8 GB USB drives embedded
VMware tools	ESX-OS data volume tool is migrated from locker partition, used for core dump volume and finally the partition is rubbed out

Fig. 3a to Fig. 3b, it can be noticed that for different iterations the starting nodes have changed to get the optimized shortest traversal route for the given parameters. For example, the starting node (marked as 1) in Fig. 3a has become node number 5 in Fig. 3b and the node number 16 in Fig. 3a has become the starting node in Fig. 3b. This happened because we have given different parameter list for EVACO algorithm for execution in these two cases. We further observe that the choice of next node changes with the change in starting node. The traversing of nodes from 1 to 20 is an implication of the fact that these virtual instances (called as nodes) are available to be allocated to the incoming sub tasks. This is possible because the idle

Fig. 3 (a) Iteration 1 and (b) Iteration 2 of EVACO algorithm

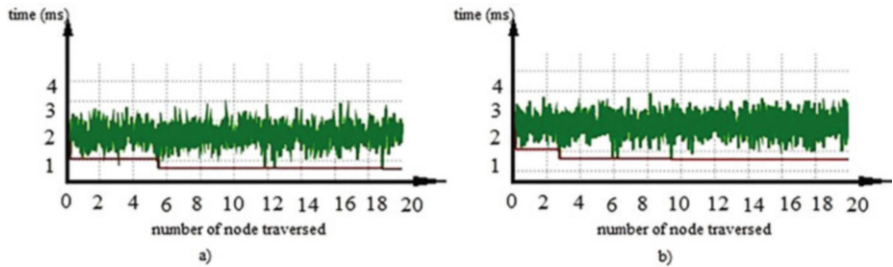
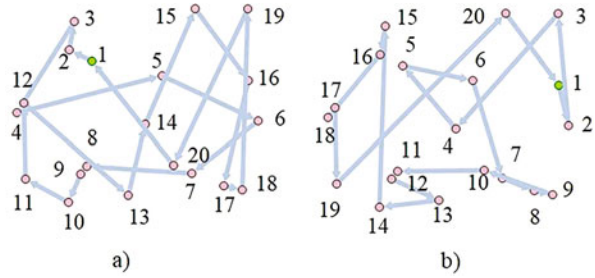


Fig. 4 Performance comparison graph of (a) Iteration 1 and (b) Iteration 2 of EVACO algorithm

index (as described in the Eq. 10) of these nodes are high and hence sub tasks are allocated to them.

The performance comparison for the stated iterations is demonstrated in Fig. 4a and Fig. 4b which reflects the time (in milliseconds) required to traverse all 20 nodes. The numeric weights of the time requirement to execute the proposed algorithm have been shown in Table 4 for further comparisons.

From Table 4, it is clearly noticeable that EVACO algorithm substantially provides better time complexity than that of other algorithms like PSO, GA, IACO, SACO, ACO etc. We have calculated mean, variance and standard deviation weights from the obtained result to compare these algorithms in terms of computational time spent for different iterations. The detailed graphical comparisons have been displayed in Figs. 5, 6, 7, 8, 9 and 10. From the generated graphs, we can claim that our proposed EVACO algorithm and the associated mathematical model stands higher than the existing approaches (Table 5).

7 Conclusion

ACO has been able to show how effectively resources can be optimized for improved allocation policy. On the other hand, virtualization proved its significance to cloud computing by reducing the infrastructure overhead and increasing the throughput of the data centres. Combined, they can be used for enhancing the

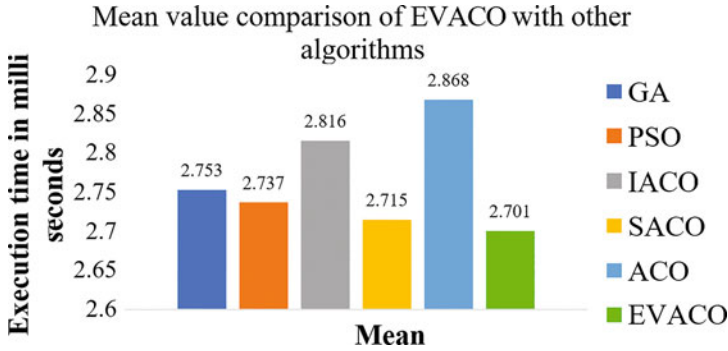


Fig. 5 Comparison of mean for PSO, GA, IACO, SACO, ACO and EVACO

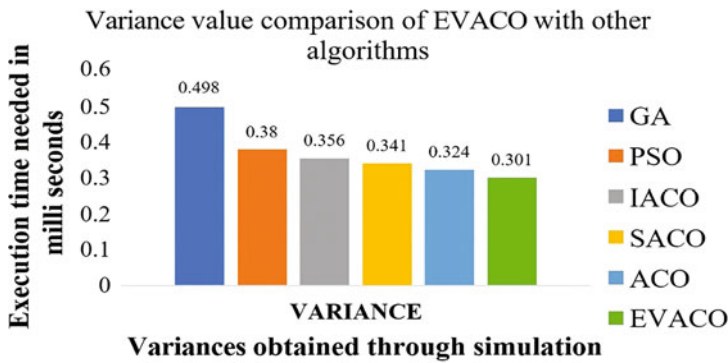


Fig. 6 Comparison of variance for PSO, GA, IACO, SACO, ACO and EVACO

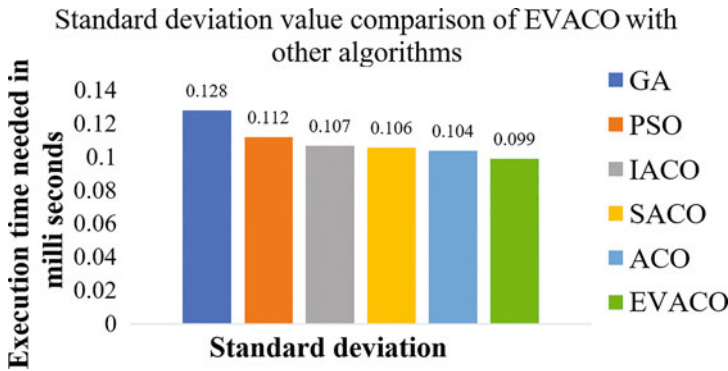


Fig. 7 Comparison of standard deviation for PSO, GA, IACO, SACO, ACO and EVACO

resource optimization policy in partially virtualized cloud platforms. In this paper, we have proposed a novel algorithm (called as EVACO) that applies ACO in a modified approach in a partially virtualized cloud platform for finding feasible

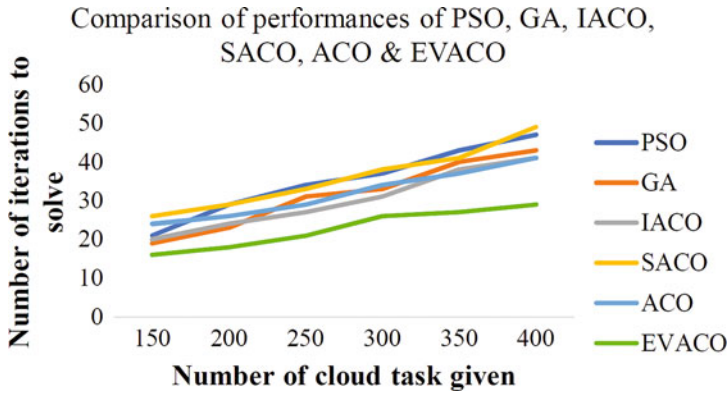


Fig. 8 Comparison of performance analysis for PSO, GA, IACO, SACO, ACO and EVACO

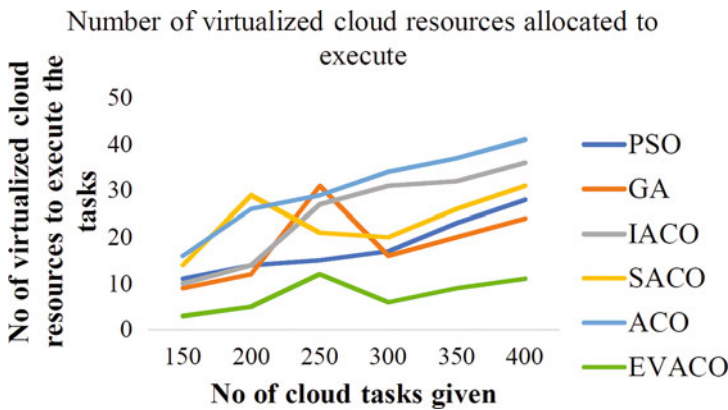


Fig. 9 Comparison of resource requirements for PSO, GA, IACO, SACO, ACO and EVACO

Fig. 10 Comparison of makespan requirements for PSO, GA, IACO, SACO, ACO and EVACO

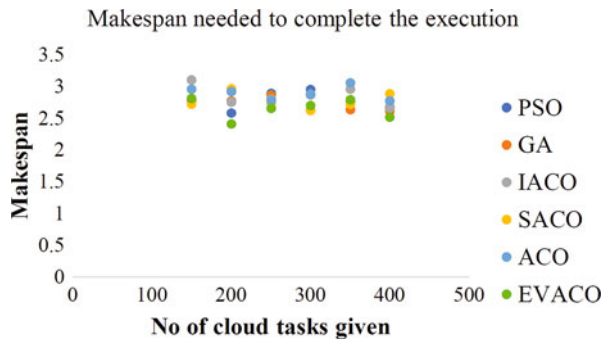


Table 5 Time comparison of EVACO vs. other algorithms for 30 iterations

#Iterations	PSO	GA	IACO	SACO	ACO	EVACO
1	2.753	2.871	2.748	2.846	2.949	2.617
2	2.862	2.510	2.800	2.943	2.730	2.503
3	2.958	2.677	2.999	2.981	3.079	2.714
4	2.660	2.675	2.823	2.861	2.926	2.666
5	2.801	2.779	2.722	2.650	2.909	2.709
6	2.603	2.619	2.791	2.797	2.814	2.699
7	2.733	2.721	2.591	2.955	2.804	2.805
8	2.688	2.882	2.820	2.917	2.744	2.635
9	2.683	2.871	2.720	2.519	2.904	2.800
10	2.888	2.685	2.963	2.520	2.906	2.741
11	2.619	2.742	2.729	2.785	2.792	2.673
12	2.748	2.619	2.519	2.578	2.720	2.720
13	2.800	2.709	2.677	2.889	2.817	2.811
14	2.999	2.835	2.675	2.952	3.030	2.943
15	2.823	3.101	2.779	2.760	2.858	2.814
16	2.722	2.752	2.619	2.669	2.961	2.638
17	2.791	2.795	2.721	2.617	2.728	2.521
18	2.591	2.714	2.882	2.503	2.940	2.606
19	2.820	2.605	2.871	2.714	2.846	2.713
20	2.720	2.706	2.685	2.823	2.943	2.507
21	2.963	2.828	2.742	2.722	2.981	2.825
22	2.729	2.707	2.619	2.791	2.861	2.639
23	2.519	2.693	2.709	2.591	2.650	2.670
24	2.520	2.754	2.835	2.820	2.797	2.598
25	2.785	2.774	3.101	2.720	2.955	2.813
26	2.578	2.768	2.752	2.963	2.917	2.409
27	2.889	2.860	2.795	2.729	2.775	2.654
28	2.952	2.629	2.862	2.619	2.879	2.701
29	2.760	2.633	2.958	2.721	3.053	2.789
30	2.669	2.611	2.660	2.882	2.769	2.513

resource allocation policy. In this context, we have also designed a new utilitarian mathematical model. This model shows how a cloud user can get the optimized solution for a given set of tasks. We have run rigorous simulations by setting up the environment using cloud analyst simulator. The simulation results shown in Table 4 explains that the performance of our proposed algorithm is quite satisfactory. We have compared our simulation results with other existing algorithms and found that our proposed algorithm is giving better results than others. Overall, through the simulation we have shown that the performance of our proposed EVACO algorithm is excellent and is better than other existing algorithms in terms of optimal resource allocation and overall execution model.

References

1. Shyam, G. K., & Chandrakar, I. (2018). Resource allocation in cloud computing using optimization techniques. In B. Mishra, H. Das, S. Dehuri, & A. Jagadev (Eds.), *Cloud computing for optimization: Foundations, applications, and challenges. Studies in Big Data* (p. 39). Springer. [10.1007/978-3-319-73676-12](https://doi.org/10.1007/978-3-319-73676-12).
2. Mishra, A. K., Umrao, B. K., & Yadav, D. K. (2018). A survey on optimal utilization of preemptible VM instances in cloud computing. *The Journal of Supercomputing*, 74, 5980–6032. <https://doi.org/10.1007/s11227-018-2509-0>
3. Choudhary, A., Gupta, I., Singh, V., & Jana, P. K. (2018). A GSA based hybrid algorithm for bi-objective workflow scheduling in cloud computing. *Future Generation Computer Systems*. <https://doi.org/10.1016/j.future.2018.01.005>
4. Guerrero, G. D., Cecilia, J. M., Llanes, A., Garca, J. M., Amos, M., & Ujald, M. (2014). Comparative evaluation of platforms for parallel Ant Colony Optimization. *The Journal of Supercomputing*, 69(1), 318–329. <https://doi.org/10.1007/s11227-014-1154-5>
5. Asgari, S., Jamali, S., Fotohi, R., et al. (2021). Performance-aware placement and chaining scheme for virtualized network functions: A particle swarm optimization approach. *The Journal of Supercomputing*. <https://doi.org/10.1007/s11227-021-03758-9>
6. Yu, J. F., Li, Y., Yu, H. S., & Shen, Q. (2019). Resources optimization deployment in collaborative manufacturing project based on adaptive Ant Colony Algorithm computer integrated manufacturing systems, 14, 3.
7. Munir, A., Koushanfar, F., Gordon-Ross, A., et al. (2013). High-performance optimizations on tiled many-core embedded systems: A matrix multiplication case study. *The Journal of Supercomputing*, 66, 431–487. <https://doi.org/10.1007/s11227-013-0916-9>
8. Li, C., Wang, C., & Luo, Y. (2020). An efficient scheduling optimization strategy for improving consistency maintenance in edge cloud environment. *The Journal of Supercomputing*. <https://doi.org/10.1007/s11227-019-03133-9>
9. Takouna, I., Sachs, K., & Meinel, C. (2014). Multiperiod robust optimization for proactive resource provisioning in virtualized data centers. *The Journal of Supercomputing*, 70, 1514–1536. <https://doi.org/10.1007/s11227-014-1246-2>
10. Tirado, F., Barrientos, R. J., Gonzalez, P., & Mora, M. (2017). Efficient exploitation of the Xeon Phi architecture for the Ant Colony Optimization (ACO) metaheuristic. *The Journal of Supercomputing*, 73(11), 5053–5070. <https://doi.org/10.1007/s11227-017-2124-5>
11. Farshin, A., & Sharifian, S. (2019). A modified knowledge-based ant colony algorithm for virtual machine placement and simultaneous routing of NFV in distributed cloud architecture. *The Journal of Supercomputing*. <https://doi.org/10.1007/s11227-019-02804-x>
12. Moon, Y., Yu, H., Gil, J. M., et al. (2017). A slave ants-based ant colony optimization algorithm for task scheduling in cloud computing environments. *Human-centric Computing and Information Sciences*, 7, 28. <https://doi.org/10.1186/s13673-017-0109-2>
13. Wang, J., Cao, J., Sherratt, R. S., & Park, J. H. (2017). An improved ant colony optimization-based approach with mobile sink for wireless sensor networks. *The Journal of Supercomputing*. <https://doi.org/10.1007/s11227-017-2115-6>
14. Mukhopadhyay, N., & Tewari, B. P. (2022). Efficient IaC-based resource allocation for virtualized cloud platforms. In I. Woungang, S. K. Dhurandher, K. K. Pattanaik, A. Verma, & P. Verma (Eds.), *Advanced network technologies and intelligent computing. ANTIC 2021. Communications in computer and information science* (Vol. 1534). Springer. https://doi.org/10.1007/978-3-030-96040-7_16
15. Mishra, S., Sahoo, M. N., Sangaiah, A. K., & Bakshi, S. (2019). Nature-inspired cost optimization for enterprise cloud systems using joint allocation of resources. *Enterprise Information Systems*, 1-23. <https://doi.org/10.1080/17517575.2019.1605001>
16. Nguyen, T. A., Min, D., & Choi, E. (2018). A comprehensive evaluation of availability and operational cost for a virtualized server system using stochastic reward nets. *The Journal of Supercomputing*, 74, 222–276. <https://doi.org/10.1007/s11227-017-2127-2>

17. Dorigo, M., & Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical Computer Science*, 344(23), 243–278.
18. Li, C., Wang, C., & Luo, Y. (2020). An efficient scheduling optimization strategy for improving consistency maintenance in edge cloud environment. *The Journal of Supercomputing*, 76, 6941–6968. <https://doi.org/10.1007/s11227-019-03133-9>

Authenticated, Secured, Intelligent and Assisted Medicine Dispensing Machine for Elderly Visual Impaired People



Soubraylu Sivakumar, D. Haritha, S. Shanmugan, Talasila Vamsidhar, and Nidumolu Venkatram

Abstract In this modern era offering medication to the Elderly Visually Impaired People (EVIP) without the involvement of medical staff is a challenging task. Due to the improvement in the latest technology and devices, offering medication to elderly people has become very important. Choosing the right device with maximum functionality is crucial. The existing pill dispenser lacks in providing medication to the right person, at the right time, the right quantity of medicine, time-bounded delivery of medicines and timely delivery of assistive messages. The proposed architecture integrates important features that support EVIP in the medication process. A biometric finger printer sensor allows the right person to access the medication kit in a secure way. The assistive technology enables the medicine to be delivered automatically to the EVIP on time (i.e., before and after food). This intelligent system checks the delivery of medicine to EVIP is taken as prescribed. An automatic message is sent to the medical entities when the weight of the pills goes below the threshold value inside the chamber. All the features are implemented in the proposed architecture through a Timely Assistance Messaging (TAM) algorithm in an integrated manner. This proposed system can be utilized in homes, hospitals and vintage-age homes by EVIP.

S. Sivakumar (✉)

Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

e-mail: sivas.postbox@gmail.com

D. Haritha · T. Vamsidhar · N. Venkatram

Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

e-mail: haritha_donavalli@kluniversity.in; talasila.vamsi@kluniversity.in; venkatram@kluniversity.in

S. Shanmugan

Department of Physics, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

e-mail: s.shanmugam1982@gmail.com

Keywords Elderly and Visually Impaired People (EVIP) · Medicine Dispensing Machine (MDM) · Timely Assistance Messaging (TAM) · Medication · Biometric · Medicine Dispenser Monitoring and Controlling System (MDMC)

1 Introduction

In the real world, [54] most people are living in urban regions. In 2014, the global population in urban has accounted for 54%. There was an increase of 20% in the urban population from 1960. The population growth rate in the urban region is expected to be 1.84%, 1.63% and 1.44% in the year ranging from 2015 to 2020, 2020 to 2025 and 2025 to 2030 respectively. In July 2018, the Agewell Foundation [16] of New Delhi conducted a survey on the aged person across 20 states of India. The report says that 23.44% of the elderly respondents were living alone in India; this is nearly one fourth of elderly Indians.

In this modern world, most family members are busy with their professional work. They have spent less time with their family due to tension and restless work. The caretakers are needed to take care of and support the aged parents in their daily activities. In urban areas supporting the parents with medical staff is found to be more expensive to adopt in real time. For example, elderly patients having heart failure needs high attention due to life threatening. In such critical emergencies, time-bounded medication is required for the elderly and visually challenged patients. The elderly people [40] of age above 60 years will have diversity in health conditions [15] than younger people.

Most elderly people's need medical treatment for their diseases [7] and illnesses. Unattended various medical treatments at regular timing may lead to various consequences, which may lead to prolonged recovery time. Therefore, proper constancy is needed in order to remember the time and dosage of the treatment. Especially elderly and visually impaired people, who forget in taking their medicine at the proper time. Poor adherence to the medication may lead to major illness. Recovering to normal life may come with a penalty in terms of time, pain and economy due to lack of proper adherence. In order to overcome these limitations, we have designed and implemented a Timely Assistance Messaging (TAM) method for visually impaired people in an economical way.

This TAM method provides timely assistance [3] messages to EVIPs based on their activity during their medication process. Special hardware is designed that includes multiple chambers for dispensing the medicine named a Medicine Dispensing Machine (MDM). It has seven chambers filled with unique tablet strips in each chamber. The chambers in the dispenser will open only when the fingerprint of EVIP matches with the stored biometric [5] images. Each chamber will open one after another based on the Medicine Schedule (MS) table. There are five tables used in this work and it is discussed in Sect. 3. In each chamber of the medical dispenser, two Ultrasonic sensors [25] (US) are used. One US outside the chamber guides the hand movement of EVIP to reach the proper chamber. Another US inside the chamber helps to determine the movement of the hand in and out of the chamber

for taking the medicine. Stepper Motor (SM) [43] is used to open and close the door of the chamber. A Load Cell (LC) is used to measure the weight of the strips before opening and after the closing of the chamber door. It helps to measure the overdose and underdose status of the medicine in each chamber.

The contribution of this paper goes with five folds.

- (a) A biometric sensor authenticates the right person to access the medication box.
- (b) The integration of hardware and software helps to deliver the medicine automatically without the involvement of the staff.
- (c) Based on the different activities performed by the EVIP in a day, different timely alerting messages are sent to the EVIP, caretaker, medical professional and pharmacist.
- (d) When there is a deviation in taking pills, an alert message is sent to intimate the overdosage or underdosage situation.
- (e) When the weight of the strips goes below a threshold level, a reminder message for refilling the strips is sent to the doctor and pharmacist.

Section 2 deals with the literature review of various pill dispensers used by the aged person. It thoroughly analyzes the various pill dispenser in terms of technology, methods, components, use and application. Section 3 discusses the proposed architecture of the MDM. This discussion contains the component used in the construction of architecture, authentication process involved and the normal operation during medication. Section 4 gives a detailed explanation of database and algorithm implementation. The experimental setup and results are discussed in Sect. 5. The conclusion and future enhancement is discussed in Sects. 6 and 7 respectively.

2 Related Work

T. Hayes et al. [20] have developed a pill box called MedTracker. This instrumental pillbox gives 7 day storage and provides automatic data collection for non-adherence and medication errors. It is a multi-compartment pillbox, which is not programmable. This device is portable for the elder person, doesn't require a user interface and provides medication information for review on demand.

S. Mukund [36] has given a medical dispenser that gives the user the facility to set timings for dispensing the multiple pills. There are two types of indications to alert the patients, one with an LED display and another with a beep sound. First, the alarm makes the remainder of taking the pills. Second, another alarm indicates the availability of the pills. This device [35]. has the facility to program for 31 days to supply 21 various medicines to the aged person.

C. McCall et al. [31] proposed a system named RMAIS for the automatic self-management and monitoring of medication. It is an RFID based adherence system, which has a built-in scale for the measurement of dosage and a motor to rotate the plate to deliver the right medicine. It also includes various medication messages like remainder to a patient and non-compliance alerts to caregivers.

B. Abbey et al. [1] introduced a pill box that can be programmed remotely to improve medication adherence. A web application was created, which gives caregivers and health professionals [53] to program and check the pill box from a remote location. Each column in the pill box is provided with wireless capability, which communicates with the mobile phone automatically. Braille numbers are added to the chamber to enhance the readability of the chamber number by the less sight person. They also used photo video evidence to infer that a medication has been made.

K. Gupta et al. [18] have proposed a design named MedAssist. This is made up of two components, a MedAssist box and a user tag. MedAssist box is an electronic device that contains the medicine to be supplied to the elderly person. It can be programmed by the caretaker through a panel based on the requirement of medication. The User tag is a small electronic component that has a transceiver for receiving and sending messages with the MedAssist. It avoids memorizing the medicine schedule and allows people to take medicine on time. It also enables rural people to take the medicine who are not able to read the medicine's name.

H. K. Wu et al. [55] have introduced a pill box with a consumption and reminder function for confirming the activity of the individual while taking the pills. A matrix bar code is printed on each medicine bag. The elderly person has to scan the bar code with a camera before and after taking the medicine. A user interface is available on the pillbox, which displays the pill details on the screen. The entire setup works well even if there is no internet service [18]. It reduces the overall implementation cost.

D. H. Mrityunjaya et al. [33] have designed a device that helps patients to take medicine on time. It prevents geriatrics from unplanned doctor or hospital visits. This device [12] provides medicine based on the schedule. A stepper motor is associated with the motion of the compartment. An alarm system is included in the device with a buzzer and LED light. Based on time, the buzzer and LED will indicate the time of medication.

A. Mondragon et al. [34] gave medikit which consists of two components, a fixed pill dispenser and a portable device that conveys the medical treatment information. The pill dispenser is composed of a mechatronics design, which helps in reducing the delay in taking the pills. The system [13] design is very simple and can be used by the patient, family members, nurse, caretakers, etc. Once the device is filled with pills and timing parameters are fixed, the device can function automatically by itself in delivering the pills easily.

The author [21] has given a pill reminder and dispenser for elderly people for taking medicine. This is easy for people to use and made a with low cost automated device. The device included in the dispenser is an IR sensor, GSM module, real time clock and LCD display apart from a regular Arduino microcontroller. The LCD displays the pill details and time of intake. The GSM module sends the message to the caretaker or family physicians about the medication. The IR sensor detects the opening of the lid of the pill box. The real time clock is used to sync the diverse activities of the medication.

J. Jennifer et al. [23] suggested a teleconference based medicine dispenser for rural patients. They have used ultrasonic sensors, temperature sensors, the heart beat

sensors, cameras, load cells and headphones in their medical machine. The machine dispatches tablets based on the conversation between the doctor and the patient. Everything is upgraded on the server. Android phone is used for notification of messages to patients. The machine [51] is named as ATM Medical Machine (AMM). The medicine is given to the patients instantly.

3 Proposed Architecture

The proposed architecture is presented in Fig. 1. MDM is a fully automated IoT [14, 29] based system to supply pills to the EVIP in a timely manner. The pills included in the architecture [19] contain medicine for a heart and diabetic elderly patient [32] and it is displayed in the Medicine DEtail Table (MDE) 1. This system is capable of giving time based information to the EVIP [26] through an Android based mobile phone. The mobile phone is kept with the EVIP, while they move to their day-to-day

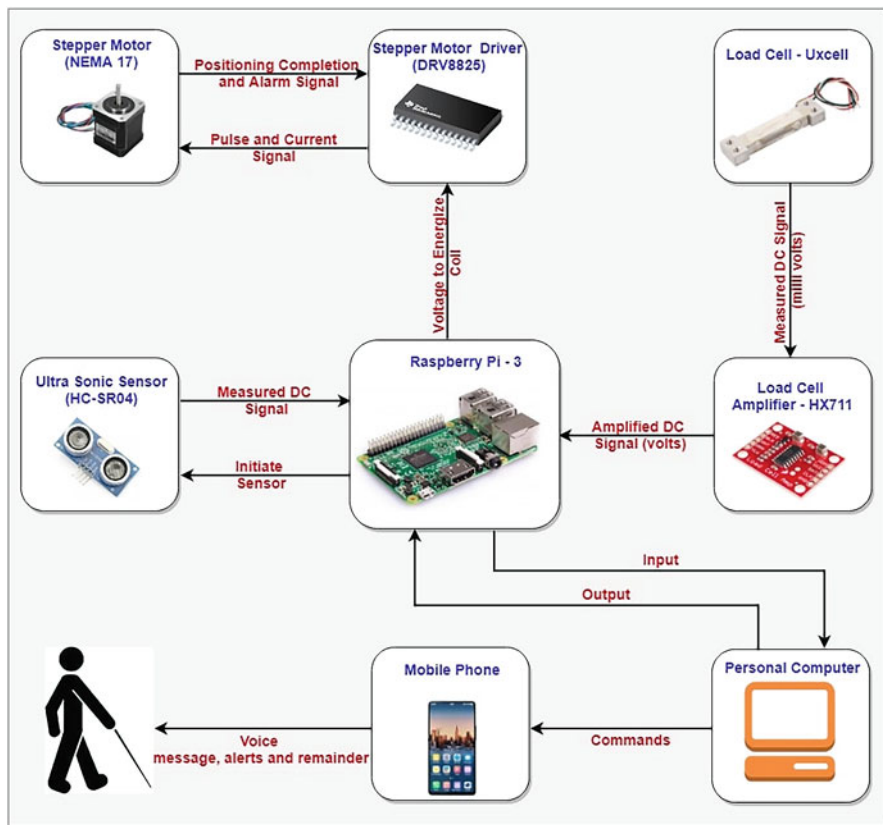


Fig. 1 Proposed architecture Medicine Dispensing Machine (MDM)

Table 1 Shows the Medicine Detail table (MDE)

Sl. no	Tablet name	Tablet weight (WT)	Tablet description	Tablet manufacturer	No. of tablets (NTS)	Purpose of the tablet
1	Imdur 30	30 mg	Prolonged Release Isosorbide-5-mononitrate Tablets B.P.	Astra Zeneca, Pharma India Limited, 12th Mile, Bellary Road, Bangalore-560083	30	It prevents angina or chest pain in patients with a certain heart condition. (coronary artery disease)
2	Clopilet	75 mg	Clopidogrel Tablets I.P.	Sun Pharma laboratories Ltd., Vill: Kolkhar, Mirza Palashbari Road, P. O. Palashbari, Dist: Kamrup, Assam: 781128	15	It avoids blood clots after little procedure and keep opens blood vessels.
3	Cardivas 6.25	6.25 mg	Carvedilal Tablets I. P.	Sun Pharma laboratories Ltd., Vill: Kolkhar, Mirza Palashbari Road, P. O. Palashbari, Dist: Kamrup, Assam: 781128	10	It is used to overcome heart failure and high blood pressure during heart attack.
4	Telma 40	40 mg	Telmisartan Tablets I.P.	Glenmark Pharmaceuticals Ltd., Village: Kishanpura, Baddi-Nalagarh Road, Tensil Baddi Distt, Solan, (H.P.) – 173205	30	It is used to prevent stroke, treats high blood pressure and other heart diseases.
5	Crosspan-DSR	40 mg	Pantoprazole Sodium with Domperidone (Sustained release) capsules	Reltsen Health Care, Spl. Plot. No: 9-11, PIPDIC Electronic Park, Thirubuvantai, Puducherry-605107	10	Stomach ulcers, Intestinal ulcers, Gastro-intestinal reflux, Treatment for symptoms associated with diabetic gastroparesis or idiopathic.
6	Rosuvas 10	10 mg	Rosuvastatin Tablets I.P.	Sun Pharma Laboratories Ltd., Plot. No:107-108, Namli Block, P. O. Ranipool, East Sikkim-737135	15	It is used in long term treatment of stroke and heart disease by reducing the bad cholesterol in the body.
7	Nexovas 10	10 mg	Cilnidipine Tablets I.P.	Macleods Pharmaceuticals Ltd, Khasra No: 21, 22, 66, 67, 68, Aho Yangtam, Nanchepung, P. O. Ranipool, Sikkim-737135	10	It is used for the management of high blood pressure by relaxing the blood vessels.

activities. The information about their medication is conveyed through the mobile phone of EVIP. The presence of a phone with the person is important for the percent implementation of the system.

Raspberry pi [45] gives the command to the mobile phone through the dongle. The command is processed in the Android phone, and it is transformed into a voice command. Based on the announcement, the EVIP [49] has to move towards the medication kit. The EVIP will be advised to provide fingerprint authentication [41] for accessing the medication kit. The biometric device [50] is present on the right hand side of the medication kit.

3.1 Dispensing Machine

3.1.1 Construction of Dispensing Machine

The MDM is made up of a wooden box. There are 7 chambers in the medication kit. Every two chambers are controlled by a Raspberry pi. The medication kit requires three Raspberry pi for complete control of the system. A chamber contains three sub chambers. Figures 2 and 3 show the front view and top view of a chamber. A chamber consists of a Load Cell (LC), Stepper Motor (SM) and Ultrasonic Sensor (US) [8] arranged from right to left as a separate subchamber. The LC subchamber consists of a plate for holding the tablet or pills, a load cell to weigh the tablet, the load cell amplifier to convert the physical force into a digital signal and an ultrasonic sensor to monitor the hand movement in and out of the chamber for taking the pills. The SM subchamber consists of a stepper motor to rotate the disk; a stainless steel lead attached to the door opens the chamber and a controller to check the rotation activities of the motor. The US subchamber consists of a sensor to find the moving hands of the EVIPs [37] across the medication kit. The entire three Raspberry

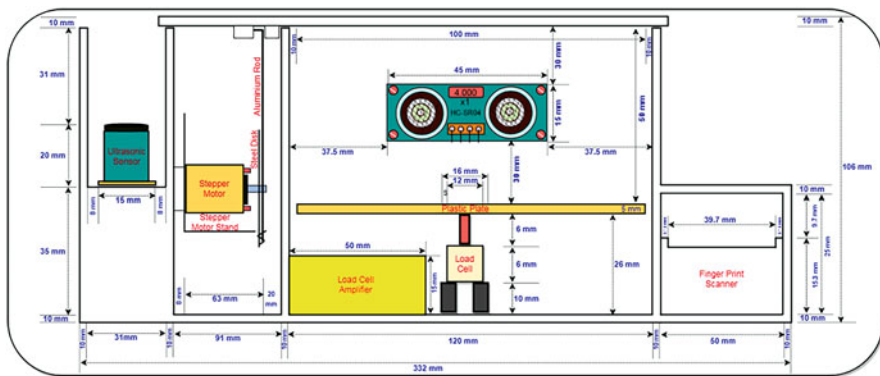


Fig. 2 Shows the arrangement of all components in a single chamber of the medication kit (front view)

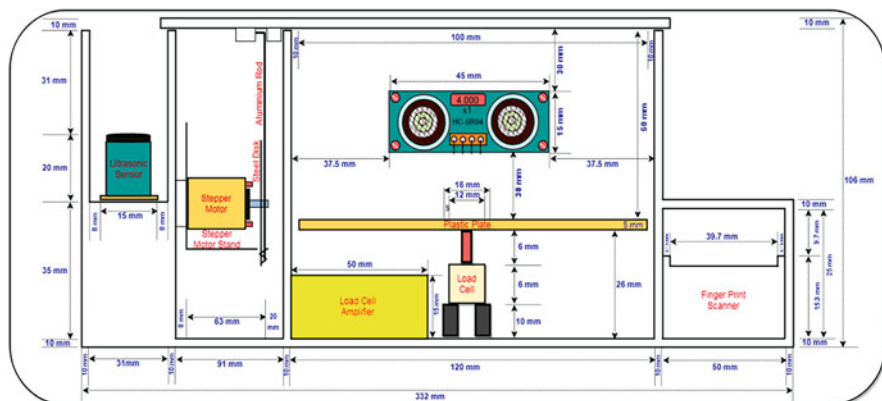


Fig. 3 Shows the arrangement of all components in a single chamber of the medication kit (top view)

pi are controlled by a Personal Computer (PC) [39] called a Medicine Dispenser Monitoring and Control (MDMC) system. The chamber number is crafted on the top of each chamber door. This number is used by the EVIP [30] to locate their respective opened chamber using the Braille method of reading.

3.2 Authentication Process

Authentication is provided for the safest delivery of the medicine to the EVIP. Unsafe delivery of the pills may lead to the incorrect pills to the EVIPs. Dispatching of incorrect pills to EVIP may lead to overdosage or health [47] related side effects. The Fig. 4 shows the entire authentication process [6] of EVIP. The fingerprint scanner captures the biometric payload and passes on the payload to MDMC through Raspberry pi. The fingerprint of the EVIP will be verified with the already stored images of the EVIP in the MDMC. If there is an exact match of the payload, the EVIP will be informed about the successful authentication [28]. If it doesn't match the stored information, the system will allow a maximum of three tries. After the third authentication failure, the MDMC will lock the opening of the medication kit. The normal operation of the system can be taken up by the caretaker.

3.2.1 Recovery of the System

When the authentication fails, the recovery of the system will be done by the caretaker of the EVIP. It is implemented in software using the password lock through the android phone [4]. The app will be installed in the mobile phone of the caretaker. During the unsuccessful authentication, the caretaker will be asked to unlock the

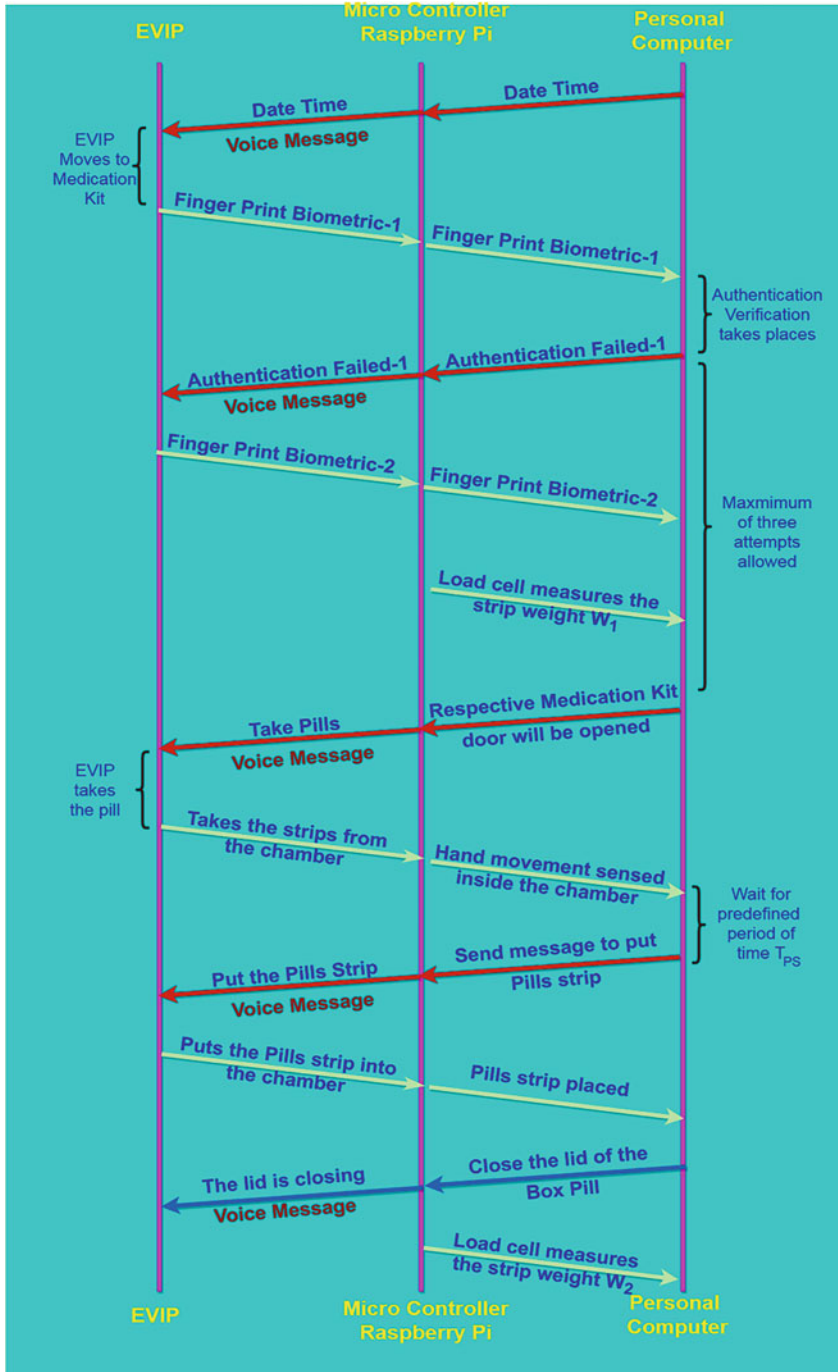


Fig. 4 Shows the sequence of steps involved during the authentication process in the opening the medication kit

Table 2 Shows the Medicine DOsage table (MDO)

Sl. no	Chamber ID	Tablet name	Tablet dosage (D _{PD})	Tablet session	Tablet time
1	C001	Imdur 30	1.0	M, N	AF
2	C002	Clopilet	1.0	A	AF
3	C003	Cardivas 6.25	0.5	M, N	AF
4	C004	Telma 40	1.0	N	BE
5	C005	Crosspan-DSR	0.5	M, N	AF
6	C006	Rosuvras 10	1.0	M, N	BE
7	C007	Nexovas 10	1.0	A	AF

M morning, *A* afternoon, *N* night, *BE* before food, *AF* after food

Table 3 Shows the Medicine Scheduling table (MS)

Sl. no	Tablet session	Time slot	Tablet time	Chamber ID
1	M	BE	09:00	C006
2		AF	09:30	C001, C003, C005
3	A	BE	12:30	
4		AF	01:00	C002, C007
5	N	BE	08:00	C004, C006
6		AF	08:30	C001, C003, C005

M morning, *A* afternoon, *N* night, *BE* before food, *AF* after food

system for its normal operation. The caretaker will make a phone call to the EVIP, to know what happens during the authentication process. If the caretaker is convinced, he/she will authorize access to the EVIP to the medication kit by providing the password. The MDMC will check the password and unlock the system [33]. The system will return back to normal operation, where the pills will be dispensed in a customary way. An announcement will be made to the EVIP to take medicine in terms of name, dosage and chamber no. based on the Medicine DOsage Table 2 (MDO).

3.3 Normal Operation

After authentication, pi refers to the Medicine Scheduling Table 3 (MS) for the opening of the chambers. Built on the day and time, the list of chambers to be opened is chosen from the schedule table. Medicine chambers which are located on the rightmost side of the biometric [48] subchamber will have high priority for opening. Each chamber will be opened only after the previously opened chamber is closed. A chamber will be closed completely only after the pill strips are kept on the plate.

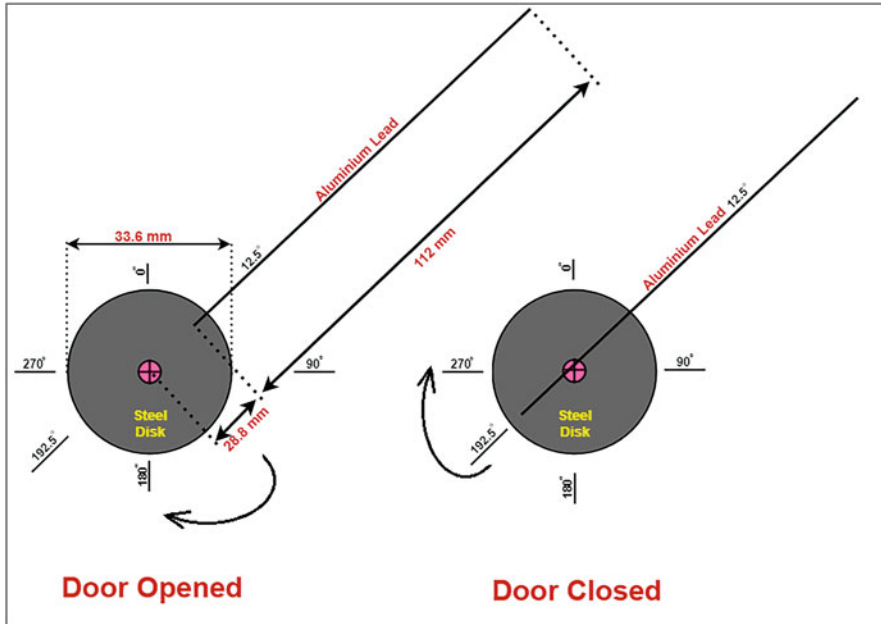


Fig. 5 Shows the position of the disk in opening and closing the door

3.3.1 Sequence of Operation for a Chamber

Based on time, the MDMC will initiate the respective Pi for the opening of the chamber door. The S.M. will rotate from 192.5° to 12.5° in a clockwise direction for the opening of the chamber. The opening and closing of the chamber are shown in Fig. 5. The Stepper motor will rotate in half of the clockwise direction. Pi will initiate the controller to give the required current pulse to the S.M. Once the door is opened, a voice message will be given to the EVIP. Based on the chamber number on the top of the door and with the help of a voice message, the EVIP can move his hand across the medication kit. While moving the hand, the U.S. in each chamber will serve to move the hand to the correct opened chamber. When the hand moves in the wrong direction, it will be sensed by the U.S. in any one of the chambers. Due to modifications in the distance, the U.S. will be informed about the movement of the hand to MDMC. In turn, the MDMC will convey the message to EVIP.

Before opening the chamber, the Pi will record the initial weight of the L. C. in milli-volts. The L. C. amplifier will scale the millivolts to volts for further measurement. The measured value will be sent to the MDMC through Pi. This value is called W_1 . Once the user has picked the right chamber, he can take the pills strip from the plate. The name of the pill and the dosage quantity along with the purpose of the pill will be informed to the EVIP by phone. After EVIP takes the pill, he/she may keep the pills strip on the plate.

Table 4 Shows the Medicine Weight Threshold table (MWT)

Sl. no	Chamber ID	Tablet name	No. of strips (N_{SC})	Total weight of a single strip (W_S) (mg)	Empty strip weight (W_{ESW}) (mg)	Threshold weight (W_{TWS}) (mg)
1	C001	Imdur 30	2	1050.0	150	270
2	C002	Clopilet	2	1325.0	200	500
3	C003	Cardivas 6.25	2	92.5	30	55
4	C004	Telma 40	1	1325.0	125	285
5	C005	Crosspan-DSR	2	445.0	45	205
6	C006	Rosuvastatin 10	2	180.0	30	70
7	C007	Nexovas 10	3	125.0	25	65

Table 5 Shows a list of phone numbers for sending alert message

Sl. no	Person	Primary number	Secondary number
1	EVIP	98*** **49	94*** **78
2	Caretaker	88*** **19	76*** **14
3	Doctor	86*** **56	63*** **18
4	Pharmacist	97*** **47	95*** **26

A U. S. is present inside the L.C. subchamber. It is situated above the plastic plate and attached to the back wall of the subchamber. It is used to identify the entry and exit of the hand inside the chamber for taking pills strip. When user has been informed to pick the strips from the chamber, U. S. will identify the entry of the hand into the chamber. There will be a change in the sound waves from the habitual reading. The hand entering the chamber deflects the sound wave, which creates a shorter traveling of waves. The time at which the hand enters the chamber will be recorded by P_i and stored in MDMC.

Likewise, when the EVIP keeps the strip inside the chamber, again the time of entry will be recorded. Now, the MDMC initiates P_i to trigger the S. M. The S. M. will rotate from 12.5° to 192.5° in clockwise direction for closing the chamber. The load cell [27] will read the weight of the strip. The measured weight will be transferred to MDMC. This measured value is called W_2 . Already measured weight (W_1) will be subtracted from the presently measured weight (W_2), which will give the difference in weight (ΔW) (1). The ΔW will be compared with the threshold weight (W_T) of the chamber. The threshold weight will be obtained from a Medicine Weight Threshold Table 4 (MWT).

$$\Delta W = W_2 - W_1 \quad (1)$$

If ΔW is greater than the W_T , there is no need for an emergency alert to be generated i.e., normal operation. Otherwise, a message will be attached to a report. At the end of the medication, the emergency report will be delivered to all the phone numbers which are stored in the phone Table 5.

4 Implementation

4.1 Database

There are four tables in this implementation. Each table is stored in the form of a database in MySQL. Table 1 named as MDE table gives a detailed description of the medicine to be taken by EVIP. Table 2 gives information about the dosage of medication taken by the EVIP. It is called an MDO table. The scheduling of the medication is given in Table 3 i.e., MS table. The threshold weight of each tablet in the chamber is maintained at the MWT table. The MDE table has information about the name of the tablet, the weight of a tablet in milligrams, the description of the tablet, and the manufacturer of the tablet, the number of tablets in a strip and the purpose of the tablet.

The MDO table holds the detail about the tablet name, chamber number in which the tablet is placed, the dosage of the tablet to be taken by the EVIP, session of the medication (Morning, Afternoon and Night) and time slot of the medication (Before/After food). The ‘M’, ‘N’ and ‘A’ in the tablet session column represents the morning, night and afternoon respectively of the tablet medication session. In the time slot column, the “AF” and “BF” stands for after food and before food time of taking the medicine. The MS table describes the tablet session, time slot, tablet time and chamber IDs to be opened. For example, in the MS table, the second row represents the morning session and after food, the chambers C001, C003 and C005 will be opened for medication. The MWT table gives the information about chamber ids of the medication box, tablet name inside the chamber, number of strips inside each chamber, empty strip weight and threshold weight of strips.

The empty strip weight (W_{ESW}) (2) is calculated by knowing the weight of a strip (W_S), no. of tablets in the strip (N_{TS}) and weight of a tablet in the strip (W_T).

$$W_{ESW} = W_S - (N_{TS} * W_T) \quad (2)$$

The threshold weight of a strip (W_{TWS}) (3) is calculated by knowing the weight of the empty strip (W_{ESW}), the weight of a tablet in the strip W_T and the number of tablets in the strip is set to four. The four indicates a message to be sent to pharmacists and doctors, once the number of tablets in the chamber goes to less than four.

$$W_{TWS} = W_{ESW} + (4 * W_T) \quad (3)$$

4.2 Algorithm

This section explains the algorithm [10] implementation of the system. The various notations used in the algorithms are listed in Table 6. Three algorithms are

Table 6 List of notations used in the algorithm

Sl. no	Notation	Description	Algorithm
1	T_A	Time at which the announcement made	Algorithm 1
2	T_{MW}	Maximum distance walking time of EVIP (2 min)	Algorithm 1
3	T_{Buff}	Buffer time to reach the dispenser (5 min)	Algorithm 1
4	T_{EB}	Time at which the EVIP done fingerprint biometric	Algorithms 1 and 2
5	T_{Th}	Maximum allowable time for biometric process	Algorithm 1
6	T_{clk}	Time of Medical Dispenser Monitoring and Controlling system	Algorithm 1
7	T_{med}	Predefined medication table	Algorithm 1
8	D_C	Current dosage of a tablet taken by EVIP	Algorithm 1
9	D_T	Tablet dosage to be taken by the EVIP	Algorithm 1
10	D_{PD}	Predefined dosage of a tablet	Algorithm 1
11	D_{PER}	Dosage percentage of a tablet (20%)	Algorithm 1
12	D_O	Over Dosage percentage of a tablet	Algorithm 1
13	D_U	Under Dosage percentage of a tablet	Algorithm 1
14	W_{CTC}^{Bef}	Current weight of the tablet strips in the chamber before medication	Algorithm 1
15	W_{CTC}^{Aft}	Current weight of the tablet strips in the chamber after medication	Algorithm 1
16	W_{TWS}	Threshold weight of the tablet strips	Algorithm 1
17	T_{BP}	Biometric processing time	Algorithm 2
18	T_{OB}	Overall biometric time	Algorithm 2
19	T_{PB}	Predefined biometric time (3 min)	Algorithm 2
20	T_{PS}	Predefined time to hold the pill strips	Algorithm 3
21	T_{PT}	Time at which the pills strip is taken	Algorithm 3
22	T_{PK}	Time at which the pills strip is kept inside	Algorithm 3
23	T_{TP}	Threshold time for taking pills	Algorithm 3

Algorithm 1 – Timely Assistance Messaging algorithm

Algorithm 2 – Biometric algorithm

Algorithm 3 – Individual Chamber Operation algorithm

implemented in Python for monitoring and reporting the activity of the EVIPs. The monitoring is done in the residence of the EVIPs. Reporting about the EVIP medication is given to the EVIP, caretaker, pharmacist and medical professional mobile number [2] in the form of voice and text messages.

Algorithm 1: Timely_Assistance_Messaging(slot)

```

1: Tmed = Get_tablet_time(slot)
2: if (Tclk == Tmed) then
3:     TA = Alert_EVIP("Reach the Medicine Dispenser")
4:     TEB = 0
5:     repeat
6:         TTh = TMW + TBuff + TA
7:         if (Tclk > TTh) then
8:             TA = Alert_EVIP("Reach the Medicine Dispenser")
9:         end if

```

```

10:         TEB =Check_biometric( )
11:     until (TEB!=0)
12:     required_medicine = taken_medicine = null
13:     Algorithm Biometric(TEB)
14:     ses = Get_session( )
15:     chamber_to_open = Get_medicine_schedule( ses, slot)
16:     chamber_len = Length(chamber_to_open)
17:     for ind = 0 to chamber_len do
18:         chamber_no = chamber_to_open[ind]
19:         WBefCTC = Read_load_cell(chamber_no)
20:         Algorithm Individual Chamber Operation(chamber_no)
21:         WAftCTC = Read_load_cell(chamber_no)
22:         DC = WBefCTC - WAftCTC
23:         DPD = Get_predefined_dosage(chamber_no)
24:         DT = DPD - WT
25:         DPER = (DT/100)*20
26:         DO = DT + DPER
27:         DU = DT - DPER
28:         if (DC > DO) then
29:             Alert_doctor("Over Dosage")
30:             Alert_caretaker("Over Dosage")
31:         end if
32:         if (DC < DU) then
33:             Alert_doctor("Under Dosage")
34:             Alert_caretaker("Under Dosage")
35:         end if
36:         WTWS = Get_threshold_weight(chamber_no)
37:         if (WCTCAft < WTWS) then
38:             Add_message(required_medicine, chamber_no)
39:             Add_message(taken_medicine, chamber_no)
40:         else
41:             Add_message(taken_medicine, chamber_no)
42:         end if
43:     end for
44:     if (required_medicine != null) then
45:         Alert_pharmacists(required_medicine)
46:     end if
47:     Alert_doctor(taken_medicine)
48: end if

```

4.2.1 Timely Assistance Messaging Algorithm (TMS)

Timely Assistance Messaging algorithm 1 controls the overall activity of the system. This algorithm takes a 'slot' argument, which identifies the time at which the medication has to be taken to i.e., (before/after food). The *Get_tablet_time()* takes 'slot' as an argument and returns the time in which the medication has to be taken from the MS table i.e., T_{Med}. It checks the time of MDMC T_{clk} is equal to T_{Med}. If it is true, it sends an alert to the EVIP by conveying him to move towards the dispenser. Information is conveyed to EVIP through *Alert_EVIP()* function, which returns a value T_A i.e, the time at which the announcement was made. The biometric

[24] time T_{EB} is initialized to zero. T_{MW} is the time taken by the EVIP to reach the medication kit by walking i.e., maximum distance (2 min). T_{Buff} is the buffer time to reach the dispenser. i.e., additional time is taken to reach the medication kit around 5 min. (due to tiredness, obstacles in the middle, went in the wrong direction or announcement message not reached properly). T_{Th} is the threshold time i.e., the maximum allowable time for the biometric process. The threshold time is calculated using the formula (4). T_{MW} and T_{Buff} are fixed values, while T_A will vary in time. Within the threshold time T_{Th} , if the EVIP has not been reached, a second announcement will be made.

$$T_{Th} = T_{MW} + T_{Buff} + T_A \quad (4)$$

When the MDMC clock exceeds the value of T_{Th} , a second announcement will be sent to the EVIP. If the MDMC clock is within the T_{Th} , then the fingerprint scanner will be checked. The *Check_biometric()* function returns 0 if the biometric is not done by the EVIP. Otherwise, it returns the time of biometrics done by the EVIP. *Biometric()* algorithm monitors the entire activity of the fingerprint scanner and it takes T_{EB} as an argument. Two variables '*required_medicine*' and '*taken_medicine*' are initialized to null. A *Get_session()* function returns the session of the day in the form of '*M*' for the morning, '*A*' for the afternoon and '*N*' for the night. The *Get_medicine_schedule()* function is called with '*ses*' and '*slot*' as arguments. This function refers to the medicine schedule table and returns the number of chambers to be opened to the variable '*chamber_to_open*'. The *Length()* function returns the no. of chambers to be opened for medication. A *for* loop is used to open all the chambers one after another in the list '*chamber_to_open*'. The *Read_load_cell()* function reads the present weight value of the strips inside the chamber. The *Individual_chamber_operation()* algorithm opens and closes the specific chamber for medication. Before opening and closing the chamber, the *Read_load_cell()* calculates the weights of the strips W_{CTC}^{Bef} and W_{CTC}^{Aft} i.e., as the current weight of the tablet strips inside the chamber before and after medication respectively.

The current dosage D_C taken by the EVIP can be calculated using the current weight of the tablet inside the chamber before and after the medication is shown in the Eq. (5).

$$D_C = W_{CTC}^{Bef} - W_{CTC}^{Aft} \quad (5)$$

The predefined dosage D_{PD} for a tablet is obtained by *Get_predefined_dosage()* function by looking at the MDO table. The tablet dosage D_T to be taken by the EVIP is calculated by using the formula (6).

$$D_T = D_{PD} * W_T \quad (6)$$

Dosage percentage D_{PER} is calculated for adjustment in errors as 20% of the tablet dosage D_T . The dosage percentage is calculated using the formula (7), it is

considered an error factor due to slight variation in the measurement of the load cell weight.

$$D_{PER} = \frac{D_T}{100} * 20 \quad (7)$$

This D_{PER} is helpful in calculating overdosage (8) and underdosage (9). The D_O overdosage is calculated as a sum of D_T and D_{PER} . The D_U under dosage is calculated as a difference between D_T and D_{PER} .

$$D_O = D_T + D_{PER} \quad (8)$$

$$D_U = D_T - D_{PER} \quad (9)$$

If the current dosage is greater than D_O , an alert message about *overdose* is sent to the doctor and the caretaker. If the current dosage is less than D_U , an alert message *underdose* is sent to the doctor and caretaker. The threshold weight W_{TWS} of a tablet strip in a chamber is obtained from the MWT table using the function *Get_threshold_weight()*. When the measured weight W_{CTC}^{Aft} is less than the W_{TWS} , 'chamber' is added to the 'required_medicine' and 'taken_medicine' lists. Otherwise, 'chamber_no' is added only to the 'taken_medicine' list. If 'required_medicine' is not equal to null, an alert message will be sent to pharmacists and doctors with the 'required_medicine' and 'taken_medicine' lists as the message. Otherwise, an alert message will be sent only to the doctor.

Algorithm 2: Biometric(TEB)

```

1: no_of_attempt = 0
2: repeat
3:   verification = Fingerprint_biometric( )
4:    $T_{OB} = T_{EB} + T_{BP}$ 
5:   if ( $T_{OB} > T_{PB}$ ) then
6:     no_of_attempt = no_of_attempt + 1
7:     Alert_EVIP("Again Do Biometric")
8:   else
9:     Alert_EVIP("Authentication Successful")
10:    Exit_Biometric( )
11:  end if
12:   $T_{EB} = \text{Check\_biometric}( )$ 
13: until ((no_of_attempts <= 3) && (verification != True))
14: if (no_of_attempts > 3) then
15:   Alert_EVIP("Authentication Failed")
16:   Alert_caretaker("Authentication Failed Unlock")
17: end if

```

4.2.2 Biometric Algorithm

The *Biometric()* algorithm alerts the successful and unsuccessful operation of the biometric process. It gives a maximum of three attempts for EVIP to access the medication kit. Once three attempts are made, the biometric system will be locked. The system can be recovered by the caretaker by unlocking the system after giving a password. The variable '*no_of_attempt*' is initialized to zero and it keep tracks of the total number of attempts made by the EVIP. The *Fingerprint_biometric()* algorithm is called for the verification of the fingerprint payload. The time taken by the *Fingerprint_biometric()* for verification is T_{BP} . This function returns true if the payload matches the '*verification*' variable. The overall biometric time T_{OB} is calculated by summing the T_{EB} and T_{BP} . The time T_{PB} is a predefined time for the biometric process. It is kept at maximum based on the processing time of the *Fingerprint_biometric()* function. If T_{OB} is greater than T_{PB} , then the number of attempts is incremented by one and a notification is sent to EVIP informing them to do authentication again. Otherwise, the authentication successful alert message is sent to the EVIP and *Biometric()* algorithm is exited. The biometric process will repeatedly check until "*no_of_attempts*" is less than three and the '*verification*' of biometrics is false. If the *no_of_attempts* is greater than three; two alert messages are sent, one for informing EVIP about the failure of the authentication process [44] and another for the informing caretaker about the failure and to unlock the process.

Algorithm 3: Individual Chamber Operation(chamber_no)

```

1: Door_open(chamber_no)
2: Alert_EVIP("Take Pills Strip")
3:  $T_{PT} = \text{Check\_inside\_US}()$ 
4:  $T_{PK} = 0$ 
5:  $T_{TP} = T_{PT} + T_{PS}$ 
6: repeat
7:    $T_{PK} = \text{Check\_inside\_US}()$ 
8:   if ( $T_{PK} > T_{TP}$ ) then
9:     Alert_EVIP("Keep the Pills Strip Inside")
10:     $T_{TP} = T_{Clk} + T_{PS}$ 
11:     $T_{PK} = 0$ 
12:   end if
13:   if ( $T_{PK} \leq T_{TP}$ ) then
14:     Alert_EVIP("The Chamber Lid is Closing")
15:     Exit_Individual_Chamber_Operation( )
16:   end if
17:until ( $T_{PK} = 0$ )

```

4.2.3 Individual Chamber Operation Algorithm (ICO)

The *Individual_Chamber_Operation()* algorithm monitors the activity of the chamber door and alerts are sent to EVIP. Each chamber is opened by *Door_open()* function, which takes the '*chamber_no*' as an argument. An alert message is sent to inform EVIP about taking the pills. The *Check_inside_US()* is a function that

checks the hand movement inside and outside of the chamber for taking and placing the pill strips. This function returns a time value T_{PT} , when the pill strip is taken and again it returns a value T_{PK} , when the pill strips are kept. Initially, the T_{PK} is initialized to zero. The time T_{PS} is the holding time of the strips for taking medicine. It is a predefined time of about 3 min. The total time that took [52] in processing the pill strip is obtained by summing the T_{PT} and T_{PS} . If T_{PK} is greater than T_{TP} , then an alert message is sent to EVIP informing them to keep the pills and a new T_{TP} is calculated by summing the current clock time T_{clk} and time to hold the strip T_{PS} . Otherwise, an alert message is sent to the EVIP, saying that the chamber is closing and the algorithm exit. The algorithm checks continue until the EVIP keeps the pill strips inside the chamber by the condition that T_{PK} is not equal to zero.

5 Result and Discussion

5.1 Experimental Setup

5.1.1 Medical Dispenser Monitoring and Controlling System (MDMC)

The system is implemented in Intel core i7 7th generation processor. It has a RAM capacity of 8GB and 1TB of hard disk. It is running on Ubuntu 16.04 operating systems. This PC controls and monitors the Raspberry Pi [17] microcontroller through a Python script. The python version employed in the experiments is 3.5. MySQL 7.0 [42] running on port 3306 is used as a database for storing the medication tables. A Med App is designed using Android 8.0 Oreo. Web interaction takes place between MDMC and the mobile app through PHP. Web scripts are designed using PHP 7.1, which runs in the Apache 2.3.1 web server in port 8000. To send a message from MDMC to the mobile phone, *Nexmo* APIs are used. For the verification of the biometric process, we have used two layer convolutional neural network [9].

The Med App receives and reads the message from MDMC. It converts it into a voice reply through *TextToSpeech* [11] API. Python scripts run the PHP web page using the *urllib2* modules. The method *request()*, *urlOpen()* and *read()* defined in *urllib2* are needed to call the PHP web pages. The experiment comprises 7 chambers in the medicine dispenser. There is 3 Raspberry pi involved in the medicine dispenser. The Raspberry pi-1 controls the fingerprint scanner, chambers 1 and 2. Chamber 3, 4 and 5 are controlled by pi-2. The remaining chamber is controlled by pi-3. Each table in this implementation contains information related to a heart and diabetic elderly patient [38]. The complete experiment is done and monitored at the residence of the patient [22] for 3 months.

In this section, many formulas are used in the calculation of the results like W_{CTC}^{Aft} , D_O , and D_U . Tables 7 and 8 are calculated dynamically to measure the above said parameters. The table is consolidated at the end of the medication session and it can be sent as a report.

Table 7 Shows the status of filling and unfilling of tablets after medication taken by the EVIP during a night session

Sl. no	Chamber ID	Tablet Time	No. of strips in the chamber (N_{SC})	Total no. of tablets in the chamber (Before med) (N_{TTC})	Current weight of the tablet strips in the chamber (Before medication) (W_{TTCbef})	Current weight of the tablet strips in the chamber (After medication) (W_{TTCaft})	Threshold weight of the tablet strips (W_{TWS})	Action to be taken
1	001	AF	02	34	1322.31	1293.470	270	No
2	002	NT	01	11	1026.17	1025.280	500	No
3	003	AF	02	13	141.34	138.129	55	No
4	004	BE	01	24	1086.46	1043.640	285	No
5	005	AF	02	17	771.24	729.460	205	No
6	006	BE	01	04	70.53	60.870	70	Filling
7	007	NT	01	08	104.86	106.180	65	No

BE before food, *AF* after food, *NT* not taken

Table 8 Shows the over dosage and under dosage status after medication taken by the EVIP during a night session

Chamber ID	Predefined dosage (D_{PD})	Tablet weight (W_T)	Dosage % (D_{PER})	Tablet dosage (D_T)	Over dosage (D_O)	Under dosage (D_U)	Current weight of the tablet strips in the chamber (Before medication) ($W_{CTC_{Bef}}$)	Current weight of the tablet strips in the chamber (After medication) ($W_{CTC_{Aft}}$)	Current dosage (D_C)	Action to be taken
01	1.0	30.00	6.000	30.000	36.000	24.00	1322.31	1293.47	28.840	CD
02	1.0	75.00	15.000	75.000	90.000	60.00	1026.17	1025.28	0.890	NT
03	0.5	6.25	0.625	3.125	3.750	2.50	141.34	138.129	03.210	CD
04	1.0	40.00	8.000	40.000	48.000	32.00	1086.46	1043.64	42.820	CD
05	0.5	40.00	4.000	20.000	24.000	16.00	771.24	729.460	41.780	OD
06	1.0	10.00	2.000	10.000	12.000	8.00	70.53	60.870	09.660	CD
07	1.0	10.00	2.000	10.000	12.000	8.00	104.86	106.18	1.320	NT

CD correct dosage, NT not taken, OD over dosage

Steps involved in the calculation of the load cell output in milligrams:

- (a) Obtain the Rated capacity in milligrams (W_{max}), Excitation voltage in volts (V_E) and Sensitivity in milli volts per volts (μ) from the specification of the load cell.
- (b) Obtain the reading of the load cell in milli volts (V_L) i.e., V_{out} .
- (c) Calculate the divider α using the formula (10).

$$\alpha = \frac{\mu}{V_E} \quad (10)$$

- (d) Calculate the full scale weight using the formula (11).

$$FSW = V_L * W_{max} \quad (11)$$

- (e) The current weight of tablets in the chamber can be calculated using the formula (12).

$$W_{CTC} = \frac{FSW}{\alpha} \quad (12)$$

The total no. of tablets in the chamber given in Eq. (13) can be calculated by using the current weight of the tablet in the chamber before medication, empty strip weight and weight of the tablet.

$$N_{TTC} = \frac{W_{CTC}^{Bef} - W_{ESW}}{W_T} \quad (13)$$

The number of strips in the chamber can be calculated using the total no. of tablets in the chamber before medication and the total no. of tablets in a strip is given in the below Eq. (14).

$$N_{SC} = Ceil \left(\frac{N_{TTC}}{N_{TS}} \right) + 1 \quad (14)$$

5.2 Result

In this section, two different scenarios are discussed (i) filling of tablet and (ii) overdose/underdose situation. All the measurements are done in milligrams (mg) in this section.

5.2.1 Scenario – Filling of Tablet

This section helps to give insights into the steps involved in calculating the weight of the strips inside the chamber. The total weight of the single strip for the tablet in chamber ‘6’ is obtained from the MWT table as 180 mg, weight of a tablet in a strip and number of tablet in a strip is obtained from the MDE table as 10 mg and 15 respectively. Using the above values, the empty strip weight W_{ESW} is calculated using Eq. (2) as follows:

$$W_{ESW} = 180 - (10 * 15) = 30 \text{ mg}$$

Using the above calculated value of empty strip weight, tablet weight taken from the MDE table (10 mg) and weight of the strips from the load cell are used in calculating the number of tablets in the chamber using the Eq. (13) as follows:

$$N_{TTC} = \frac{(70.53 - 30)}{10} = 4.053 \simeq 4$$

With the help of the above calculated value of N_{TTC} and the total number of tablets in a strip N_{TS} taken from the MDE table are used in calculating N_{SC} using the Eq. (14).

$$N_{SC} = \text{Ceil} \left(\frac{4}{15} \right) + 1 = 0 + 1 = 1$$

The weight of the strip in chamber ‘6’ is obtained with the help of load cells before and after the medication is given as 70.53 mg and 60.87 mg. In the final step, the threshold weight of the strip is calculated using the Eq. (3) with the value of W_{ESW} and W_T as 30 mg and 10 mg respectively.

$$W_{TWS} = 30 + 4 * 10 = 70 \text{ mg}$$

To make a final decision, threshold weight WT and strip weight after medication are compared. In the comparison, W_{TWS} is greater than W_{CTC}^{Aft} . So, a filling message is sent to the pharmacist and to the doctor.

In this Fig. 6, the x-axis represents the chamber ID from 1 to 7 and the y-axis represents dosage in milligrams from 0 to 1400. In each chamber along the x-axis, one line and the star is provided with different colors. The lower line represents the threshold level of each chamber and star represents the weight of the strip after medication. If we carefully notice the brown color star in chamber ‘6’ is below the threshold line 70 mg. As a result of crossing the threshold level, a filling message is sent to the medical entities and it is labeled as (6, 60.87) with a brown color star.

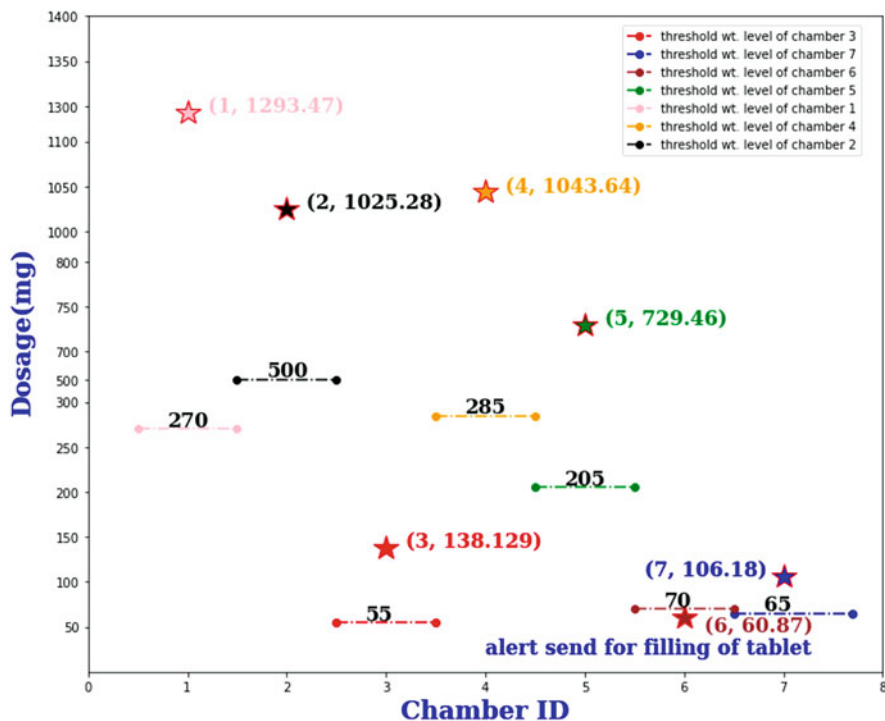


Fig. 6 Illustrates the filling of tablets when the weight of strips goes below the threshold weight in chamber 6

5.2.2 Scenario – Overdosage Situation

Table 8 lists the various parameters related to the medication taken by the EVIP after a night session. In the table, row five includes the medication status of chamber ‘5’ from the medication dispenser. The tablet dosage D_T is calculated with the values from the MDO table – Predefined dosage D_{PD} (0.5) and MDE table – Tablet weight W_T (40 mg). It is calculated as given below:

$$D_T = 0.4 * 40 = 20 \text{ mg}$$

A dosage percentage D_{PER} is calculated using D_T (tablet dosage) in order to adjust the slight variation in the measurement of the load cell and it is given below

$$D_{PER} = \frac{20}{100} * 20 = 4$$

The overdosage D_O limit and underdosage D_U limit helps to identify the status of the medication that is under control. It is calculated as given below for the given

scenario.

$$D_O = 20 + 4 = 24$$

$$D_U = 20 - 4 = 16$$

Using the load cell, the current weight of the strip inside the chamber ‘5’ before and after medication is given as $W_{CTC}^{Bef} = 771.24$ and $W_{CTC}^{Aft} = 729.46$. These values are used in calculating the current dosage as given below

$$D_C = 771.24 - 729.46 = 41.78$$

Since, $D_C > D_O$, an alert message is sent to the medical entities indicating that the EVIP has taken overdose of the tablet from chamber ‘5’. This helps to get immediate responses from the medical entities.

In Fig. 7, the x-axis represents the chamber ID and the y-axis represents the dosage in milligrams (mg). In each chamber, there are two lines and one star is provided in the figure with different colors. The lower line in each chamber represents the under limit dosage, while upper limit represents the over limit dosage.

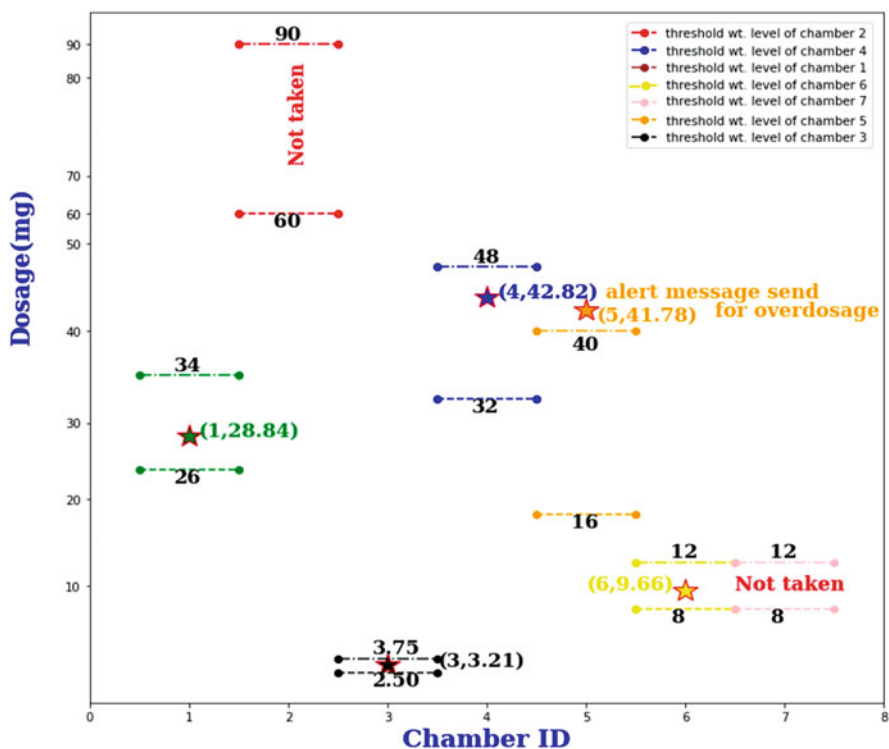


Fig. 7 Illustration of over dosage situation in chamber 5 after medication during night session

The star indicates the current dosage. Since the star has crossed the overdosage limit in Fig. 7 and it is pointed with a label (5, 41.78). The orange color in chamber 5 represents the overdosage situation.

5.3 Discussion

The prevailing medical dispenser lacks in the following (i) supplying medicine before and after food, (ii) overdosage and underdosage situation when the tablet is taken and (iii) when the weight of the pills goes below the threshold. In all these cases, no messages are sent to the medical entities to convey the situation. The proposed work has given five significant [46] capabilities to EVIP which isn't present in any of the existing pill dispensers. The `Timely_Assistance_Messaging()` algorithm controls and monitors the overall activities of the EVIP. This algorithm checks the reachability of EVIP towards medication within the predefined amount of time. When the person doesn't reach the dispenser it gives a timely alert. This calls the `Biometric()` algorithm to examine whether the right person is accessing the box. Complete medication details are obtained and the respective chambers are opened one after another for medication. The weight of the tablet, the opening and the closing of the chamber are monitored by the `Individual_Chamber_Operation()`. This function serves to monitor the overdosage and underdosage of the EVIPs medication by sensing the weight of the strips. `Individual_Chamber_Operation()` check the hand movement for taking and keeping the strips inside the chamber. It also alerts the EVIP when the strips are not placed inside the chamber for some time. When the number of tablets in strips goes below a threshold level; an alert message will be sent to the medical entities.

6 Conclusion

The complete experiment is done at the residence of the patient for the duration of 3 months. The implementation gave a promising result, in which the EVIP was given timely medication. It gave a triangle integrated relationship between the caretaker, medical professionals like doctors, pharmacists, etc. and EVIP. All three medical entities are time bounded. There is no need to continue monitoring the medication of the EVIPs. The fingerprint biometric in this system gives the right person to access the machine [55]. It doesn't need the involvement of clinical staff or nurses to deliver the medicines. It is completely assisted by the TAM algorithm. During a medication process, underdosage or overdosage messages will be sent to the medical entities. Refilling of strips alert will be sent to the pharmacists based on the weight in each chamber. TAM gives a timely alert to the right person based on the situation. This implementation gives proof of life to the medical professionals and caretakers about the EVIPs survival. It acts as a supporting tool in the field of Geriatrics.

7 Future Enhancements

The medication time of the EVIPs can be recorded and a summarized report can be sent to the medical professionals. A camera can be used at the backside of the door, which ensures the correct strips are placed in the chamber. This design can be changed to support the weekdays in the implementation of medication. This architecture doesn't deal with power failure. Giving medication to the elderly person needs a timely and critical service that runs 24/7. In this contemporary society, providing broadband internet service to the urban area for this decisive health care system is a dispute. The fall detection system of elderly people needs daily monitoring of their activities. This is not addressed in this work. A security layer can be added for the communication that takes place between the medical entities. It improves the privacy and reliability of communication.

References

1. Abbey B., et al. (2012). A remotely programmable smart pillbox for enhancing medication adherence. In *2012 25th IEEE international symposium on computer-based medical systems (CBMS)*, pp. 1–4. <https://doi.org/10.1109/CBMS.2012.6266350>
2. Arokianathan, P., Dinesh, V., Elamaran, B., Veluchamy, M., & Soubraylu, S. (2019). Automated toll booth and theft detection system. In *IEEE technological innovation in ICT for agriculture and rural development (TIAR)*, pp. 84–88. <https://doi.org/10.1109/TIAR.2017.8273691>
3. Azzabi, O., Njima, C. B., & Messaoud, H. (2017). New approach of diagnosis by timed automata. *International Journal of Ambient Computing and Intelligence*, 8(3), 76–93. <https://doi.org/10.4018/IJACI.2017070105>
4. Brindha, S., Deepalakshmi, D., Dhivya, T., Arul, U., Sivakumar, S., & Kannan, K. N. (2017). ISCAP: Intelligent and smart cryptosystem in android phone. In *2017 International conference on power and embedded drive control (ICPEDC)*, pp. 453–458. <https://doi.org/10.1109/ICPEDC.2017.8081132>
5. Chaki, J., Dey, N., Shi, F., & Sherratt, R. S. (2019). Pattern mining approaches used in sensor-based biometric recognition: A review. *IEEE Sensors Journal*, 19(10), 3569–3580. <https://doi.org/10.1109/JSEN.2019.2894972>
6. Chandrakar, P. (2019). A secure remote user authentication protocol for healthcare monitoring using wireless medical sensor networks. *International Journal of Ambient Computing and Intelligence*, 10(1), 96–116. <https://doi.org/10.4018/IJACI.2019010106>
7. Changala, R., & Rajeswara Rao, D. (2019). Development of predictive model for medical domains to predict chronic diseases (diabetes) using machine learning algorithms and classification technique. *ARNP Journal of Engineering and Applied Sciences*, 14(6), 1202–1212.
8. Chintala, R. R., Narasinga Rao, M., & Venkateswarlu, S. (2019). Performance metrics and energy evaluation of a lightweight block cipher in human sensors network. *Journal of Advanced Trends in Computer Science and Engineering*, 8(4), 1487–1490. <https://doi.org/10.30534/ijatcse/2019/69842019>
9. Chittajallu, S. M., Lakshmi Deepthi Mandalaneni, N., Parasa, D., & Bano, S. (2019). Classification of binary fracture using CNN. In *2019 global conference for advancement in technology (GCAT)*, pp. 1–5. <https://doi.org/10.1109/GCAT47503.2019.8978468>

10. Chitturi, S., Marella, S. T., Chitturi, S., & Ahammad, S. (2019). A novel cloud partitioning algorithm using load balancing in public cloud environment. *Journal of Advanced Trends in Computer Science and Engineering*, 8(7), 856–860.
11. Developers, A. (2018). *TextToSpeech API*. <https://developer.android.com/reference/android/speech/tts/TextToSpeech>
12. Dey, N., Ashour, A. S., Fong, S. J., & Bhatt, C. (2019). *Wearable and implantable medical devices: Applications and challenges*. Academic. <https://doi.org/10.1016/C2017-0-03249-4>
13. Dey, N., & Mukherjee, A. (2018). *Embedded systems and robotics with open source tools*. CRC Press. <https://doi.org/10.1201/b19730>
14. Divya, K., Kumar, V., Sujana, J. R., & Kumar, J. T. (2019). Enhanced IoT accessing security. *International Journal of Engineering and Advanced Technology (IJEAT)*, 8(4), 1586–1590.
15. Fawagreh, K., & Gaber, M. M. (2020). Resource-efficient fast prediction in healthcare data analytics: A pruned random forest regression approach. *Computing*, 102, 1187–1198. <https://doi.org/10.1007/s00607-019-00785-6>
16. Foundation, A. (2018). *Agewell studies*. https://www.agewellfoundation.org/?page_id=1162
17. Gopi, G., Priyanka, P., Bharathi, K., Rajasekhar, J., & Kommuri, K. (2019). Access control of door using face recognition and home security alert using raspberry pi and internet. *International Journal of Innovative Technology and Exploring Engineering*, 7(6C2), 68–72.
18. Gupta, K., Jain, A., Vardhan, P. H., Singh, S., Amber, A., & Sethi, A. (2014). MedAssist: Automated Medication Kit. In *2014 Texas Instruments India Educators' conference (THIEC)*, pp. 93–99. <https://doi.org/10.1109/THIEC.2014.024>
19. Harita, U., & Sagar, K. V. D. (2018). A survey on secured internet of things architecture. *International Journal of Engineering and Technology*, 7(2), 274–276. <https://doi.org/10.14419/ijet.v7i2.7.10596>
20. Hayes, T., Hunt, J. M., Adami, A., & Kaye, J. (2006). An electronic pillbox for continuous monitoring of medication adherence. *IEEE Engineering in Medicine and Biology Society Conference*, 1, 6400–6403. <https://doi.org/10.1109/IEMBS.2006.260367>
21. Jabeena, A., Sahu, A. K., Roy R., & Basha, N. S. (2017). Automatic pill reminder for easy supervision. In *2017 International conference on intelligent sustainable systems (ICISS)*, pp. 630–637. <https://doi.org/10.1109/ISS1.2017.8389315>
22. Jain, A., & Bhatnagar, V. (2017). Concoction of ambient intelligence and big data for better patient ministration services. *International Journal of Ambient Computing and Intelligence*, 8(4), 19–30. <https://doi.org/10.4018/IJACI.2017100102>
23. Jennifer, J., Marrison, M. N., Seetha, J., Sivakumar, S., & Saravanan, P. (2017). Dmmra: Dynamic medical machine for remote areas. In *2017 International conference on power and embedded drive control (ICPEDC)*, pp. 467–471. <https://doi.org/10.1109/ICPEDC.2017.8081135>
24. Joseph, T., Kalaiselvan, S. A., Aswathy, S. U., Radhakrishnan, R., & Shamna, A. R. (2020). A multimodal biometric authentication scheme based on feature fusion for improving security in cloud environment. *Journal of Ambient Intelligence and Humanized Computing*, 1-9. <https://doi.org/10.1007/s12652-020-02184-8>
25. Kamatchi, K., Sangeetha, R., Subha, V., Sivakumar, S., & Ramachandran, R. (2017). D2cmus: Detritus to cinder conversion and managing through ultrasonic sensor. In *2017 Third international conference on science technology engineering and management (ICONSTEM)*, pp. 38–43. <https://doi.org/10.1109/ICONSTEM.2017.8261254>
26. Kasthuri, R., Nivetha, B., Shabana, S., Veluchamy, M., & Soubraylu, S. (2017). Smart device for visually impaired people. In *International conference on science technology engineering and management (ICONSTEM)*, pp. 54–59. <https://doi.org/10.1109/ICONSTEM.2017.8261257>
27. Kaveeya, G. S., Gomathi, S., Kavipriya, K., Selvi, A. K. & Sivakumar, S. (2017). Automated unified system for LPG using load sensor. In *2017 International conference on power and embedded drive control (ICPEDC)*, pp. 459–462. <https://doi.org/10.1109/ICPEDC.2017.8081133>

28. Krishna, G., Kumar, P. S. P., Sreenivasa Ravi, K., Sravanthi, D., & Likhitha, N. (2019). Smart home authentication and security with IoT using face recognition. *International Journal of Recent Technology and Engineering*, 7(6), 691–696.
29. Kumar Dabbakuti, J. R. K., & Bhupati, C. (2019). Ionospheric monitoring system based on the internet of things with thingspeak. *Astrophysics and Space Science*, 264(8). <https://doi.org/10.1007/s10509-019-3630-0>
30. Liouane, Z., Lemlouma, T., Roose, P., Weis, F., & Messaoud, H. (2020). An intelligent knowledge system for designing, modeling, and recognizing the behaviour of elderly people in smart space. *Journal of Ambient Intelligence and Humanized Computing*, 1-17. <https://doi.org/10.1007/s12652-020-01876-5>
31. McCall, C., Maynes, B., Zou, C., & Zhang, N. J. (2012). An automatic medication self management and monitoring system for independently living patients. *Medical Engineering and Physics*, 35. <https://doi.org/10.1016/j.medengphy.2012.06.018>
32. Mohammadi, M., Omar, M., & Bouabdallah, A. (2018). Secure and lightweight remote patient authentication scheme with biometric inputs for mobile healthcare environments. *Journal of Ambient Intelligence and Humanized Computing*, 9, 1527–1539. <https://doi.org/10.1007/s12652-017-0574-5>
33. Mrityunjaya, D. H., Kartik, J., Uttarkar Teja, B., & Hiremath, K. (2016). Automatic pill dispenser. *International Journal of Advanced Research in Computer and Communication Engineering*, 5(7), 543–547. <https://doi.org/10.17148/IJARCC.2016.57107>
34. Mondragon, A., De Hoyos, A., Trejo, A., Gonzalez, M., & Ponce, H. (2017). Medi-kit: Developing a solution to improve attention on medical treatment. *IEEE Mexican Humanitarian Technology Conference (MHTC)*, 560, 28–33. <https://doi.org/10.1109/MHTC.2017.7926207>
35. Mukherjee, A., & Dey, N. (2019). *Smart computing with open source platforms*. CRC Press. <https://doi.org/10.1201/9781351120340>
36. Mukund, S. (2012). Design of automatic medication dispenser. *Computer Science Conference Proceedings (CSCP)*, 2, 251–257. <https://doi.org/10.5121/csit.2012.2324>
37. Najjar, M., Courtemanche, F., Hamam, H., Dion, A., & Bauchet, J. (2009). Intelligent recognition of activities of daily living for assisting memory and/or cognitively impaired elders in smart homes. *International Journal of Ambient Computing and Intelligence*, 1(4), 46–62. <https://doi.org/10.4018/jaci.2009062204>
38. Ortiz, A., Del Puy Carretero, M., Oyarzun, D., Yanguas, J. J., Buiza, C., Gonzalez, M. F., & Etxeberria, I. (2007). Elderly users in ambient intelligence: Does an avatar improve the interaction? In C. Stephanidis & M. Pieper (Eds.), *Universal access in ambient intelligence environments* (Vol. 4397). Springer. https://doi.org/10.1007/978-3-540-71025-7_8
39. Ose Aguilar, J. M., Mendonca, M., & Sanchez, M. (2020). Performance analysis of the ubiquitous and emergent properties of an autonomic reflective middleware for smart cities. *Computing*, 102, 2199–2228. <https://doi.org/10.1007/s00607-020-00799-5>
40. Panday, R., & Kumar, P. (2020). Quality of life among elderly living in old age home: A brief overview. *International Journal of Community Medicine and Public Health*, 7(3), 1123–1126. <https://doi.org/10.18203/2394-6040.ijcmph20200978>
41. Ragoo, M. (2016). Using a fingerprint-access medication cabinet to improve efficiency in an emergency department. *Emergency Medicine Journal*, 33(12), 926. <https://doi.org/10.1136/emjmed-2016-206402.49>. arXiv: <https://emj.bmj.com/content/33/12/926.2.full.pdf>
42. Raj, R. T., Sanjay, S., & Sivakumar, S. (2016). Digital licence mv. In *2016 international conference on wireless communications, signal processing and networking (WiSPNET)*, pp. 1277–1280. <https://doi.org/10.1109/WiSPNET.2016.7566342>
43. Sagar, S., Kumar, G., Xavier, L., Soubraylu, S., & Durai, R. (2017). Sisfat: Smart irrigation system with flood avoidance technique. In *International conference on science technology engineering and management (ICONSTEM)*, pp. 28–33. <https://doi.org/10.1109/ICONSTEM.2017.8261252>
44. Sahoo, S., Mohanty, S., & Majhi, B. (2020). A secure three factor based authentication scheme for health care systems using IoT enabled device. *Journal of Ambient Intelligence and Humanized Computing*, 1–16. <https://doi.org/10.1007/s12652-020-02213-6>

45. Saranu, P. N., Abirami, G., Sivakumar, S., Ramesh, K. M., Arul, U., & Seetha, J. (2018). Theft detection system using PIR sensor. In *2018 4th International conference on electrical energy systems (ICEES)*, pp. 656–660. <https://doi.org/10.1109/ICEES.2018.8443215>
46. Sarkar, M., Banerjee, S., Badr, Y., & Sangaiah, A. K. (2017). Configuring a trusted cloud service model for smart city exploration using hybrid intelligence. *International Journal of Ambient Computing and Intelligence*, *8*(3), 1–21. <https://doi.org/10.4018/IJACI.2017070101>
47. Satti, F. A., Ali, T., Hussain, J., Khan, W. A., Khattak, A. M., & Lee, S. (2020). Ubiquitous health problem (UHPr): A big data curation platform for supporting health data interoperability. *Computing*, *102*, 2409–2444. <https://doi.org/10.1007/s00607-020-00837-2>
48. Soniya, V., Sri, R., Titty, K., Ramakrishnan, R., & Sivakumar, S. (2017). Attendance automation using face recognition biometric authentication. In *2017 International conference on power and embedded drive control (ICPEDC)*, pp. 122–127. <https://doi.org/10.1109/ICPEDC.2017.8081072>
49. Soubraylu, S., Ratnavel, R., Kolla, P., Bhanu, B., Rajesh, K., & Chinnasamy, K. (2019). Virtual vision architecture for vip in ubiquitous computing. In S. Paiva (Ed.), *Technological trends in improved mobility of the visually impaired* (EAI/Springer innovations in communication and computing) (Vol. 590, pp. 145–179). Springer. <https://doi.org/10.1007/978-3-030-16450-8>
50. Tistarelli, M., & Dey, B. J. (2011). Biometrics in ambient intelligence. *Ambient Intelligence and Humanized Computing*, *2*, 113–126. <https://doi.org/10.1007/s12652-010-0033-z>
51. Ugljanin, E., Kajan, E., Maamar, Z., Asim, M., & Buregio, V. (2020). Immersing citizens and things into smart cities: A social machine-based and data artifact-driven approach. *Computing*, *102*, 1567–1586. <https://doi.org/10.1007/s00607-019-00774-9>
52. Vitabile, S., Conti, V., Collotta, M., Scata, G., Andolina, S., Gentile, A., & Sorbello, F. (2013). A real-time network architecture for biometric data delivery in ambient intelligence. *Journal of Ambient Intelligence and Humanized Computing*, *4*, 303–321. <https://doi.org/10.1007/s12652-011-0104-9>
53. Walsh, M. J., Barton, J., O’Flynn, B., Hayes, M. J., O’Mathuna, S. C., & Alavi, S. M. M. (2011). An antiwindup approach to power controller switching in an ambient healthcare network. *International Journal of Ambient Computing and Intelligence*, *3*(2), 35–55. <https://doi.org/10.4018/jaci.2011040103>
54. WHO, Global Health Observatory (GHO) data. (2020). https://www.who.int/gho/urban_health/situation_trends/urban_population_growth_text/en/. Accessed 3 Sept 2020.
55. Wu, H. K., Wong, C. M., Liu, P. H., Peng, S.P., Wang, X.C., Lin, C. H., & Tu, K.H. (2015). A smart pill box with remind and consumption confirmation functions. In *2015 IEEE 4th global conference on consumer electronics (GCCE)*, pp. 658–659. <https://doi.org/10.1109/GCCE.2015.7398716>

Prediction of Liver Disease Using Soft Computing and Data Science Approaches



Dilip Kumar Choubey , Pragati Dubey, Babul P. Tewari ,
Mukesh Ojha , and Jitendra Kumar 

Abstract The liver is the most important and one of the largest organs in the body. The Liver serves a number of functions, making it vital to the human body. When the Liver's regular functions are disrupted, it becomes a disrupted Liver. The number of liver patients has been significantly growing in recent years, making improved liver disease detection a challenging aspect of health care. The use of an automated diagnostic system can assist in identifying liver disease and improve diagnostic accuracy. As a consequence, we have used machine learning classification techniques like Logistic Regression, Gaussian Naïve Bayes, Stochastic Gradient Descent, KNN, Decision Tree, Random Forest, and SVM to design a more accurate diagnostic model. All these algorithms have implemented on the ILPD dataset with the intension to reduce time of diagnosis and earlier prediction of disease. In Future the accuracy of classification will be enhanced by feature extraction and the big dataset can also be examined for training the model and determining algorithms.

Keywords Machine learning · SVM · Logistic regression · Gaussian Naïve Bayes · KNN · Stochastic gradient descent · Random Forest · DT

D. K. Choubey (✉) · B. P. Tewari
Department of Computer Science & Engineering, Indian Institute of Information Technology
Bhagalpur, Bhagalpur, Bihar, India
e-mail: dkchoubey.cse@iiitbh.ac.in; bptewari.cse@iiitbh.ac.in

P. Dubey
Department of Bioinformatics, School of Earth, Biological and Environmental Sciences, Central
University of South Bihar, Gaya, Bihar, India

M. Ojha
Greater Noida Institute of Technology, Greater Noida, UP, India

J. Kumar
School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu, India
e-mail: jitendra.kumar@vit.ac.in

1 Introduction

Liver is the most essential and one of the largest Organ of the body. Liver has many functions including detoxifying the chemical substances, making proteins by hepatocyte cells, making triglycerides, cholesterol, Bile (required in digestion of foods), generates blood clotting components and stores glucose as glycogen. When Liver fails to perform its normal function, then it turns into a disordered Liver. Liver Disorder can be happened through variety of factors such as Viruses, alcohols, toxins, obesity, genetic inheritance, high cholesterol, and type 2 diabetes. Liver undergoes changes after exposure of these factors, The changes can be inflammatory (lead to Hepatitis), fatty accumulation or degeneration (lead to Steatosis) and both inflammatory and fatty changes (lead to Steatohepatitis). There are four stages of Liver Disorder Healthy liver, Fibrosis, Cirrhosis and Cancer. The most common disease caused by virus in Liver is Hepatitis causes inflammation in the Liver. Liver disease caused by immune system abnormality leads to autoimmune hepatitis which happens when immune cells target the Liver instead of pathogens, PBC (Primary Biliary Cholangitis) which causes destruction of bile duct in the Liver and PSC (Primary Sclerosing Cholangitis) causes inflammation and scarring of the bile ducts. Genetic causes of Liver disease leads to Hemochromatosis (excessive amount of iron is absorbed by the Liver), Wilson's disease (biliary copper excretion and pathological copper accumulation) and Alpha-1 antitrypsin deficiency an autosomal recessive disease, in which disease alpha-1 antitrypsin protein travel from Liver by blood capillaries to protect our Lungs and other body organs. If the protein has an inappropriate shape, then it stuck in the Liver and causes cirrhosis. Fatty Liver is a condition in which fats build up in the Liver, causes for Fatty Liver condition are consumption of alcohol and few medicines, obesity, diabetes, hypertension. There are two types fatty liver disorder one is Alcoholic Fatty Liver Disease occurs due to consumption of excessive amount of alcohol and another is NAFLD occurs when fatty tissue builds up in the Liver without consumption of alcohol. In maximum cases fatty liver does not cause any problem to patient but in few patient fats which is stored in the hepatocytes gives swelling or inflammation in the cell, this condition is known as NASH (Non-Alcoholic Steatohepatitis). NASH is the severe form of NAFLD, NASH leads to complications such as Liver Cancer, Liver Failure, Cirrhosis, cardiovascular disease. Diagnosis of Liver disease can be performed by Liver function test, Ultrasound, Liver biopsy, FIBROSCAN, blood test etc. Huge amount of data is immersed from the health care sector causes strain in management of records so overcome from such kind of issues we need machine-based management system. Machine Learning is the subset of AI which allows user to provide computer algorithm and immense amount of data and have the computer analyse and make decision based on input data. By analysing thousands of records, machine learning algorithms can identify patterns linked to diseases and medical conditions in the healthcare industry. There are three kinds of application in machine learning: supervised, unsupervised and reinforcement learning. We have chosen supervised machine learning classification techniques because our dataset

is labelled. In our work we used a liver patient dataset and applied seven kind of machine learning algorithms to obtain better diagnostic of Liver Disease. Logistic Regression, Gaussian Naïve Bayes, Random Forest, Stochastic Gradient Descent, K-Nearest Neighbour, Decision Tree, Random Forest and Support Vector Machine classification algorithms have been used in this work to achieve a better and accurate way for detection of Liver Disease.

Majorly all used algorithms were implemented earlier and fewer algorithms, such as Gaussian Naïve Bayes and Stochastic Gradient Descent, have been used for the first time for the prediction of diseases, which have performed very well. All these implemented algorithms have been compared with the existing algorithms for the Indian Liver Patient Dataset (ILPD).

The remaining sections of the chapter could be arranged as follows: Sect. 2 would be devoted to Motivation; Sect. 3 would be devoted to Literature Review; Sect. 4 would be devoted to Materials and Methods; Sect. 5 would be devoted to Experimental Results and Discussion; and Sect. 6 would be devoted to Conclusion and Future Scope.

2 Motivation

Liver disease is emerging rapidly with the time and causing serious illness. Around two million deaths per year worldwide are caused by liver disorders, one million due to Cirrhosis and one million due to viral hepatitis. Approximately two billion peoples consume alcohol and 75 million people are diagnosed with AFLD. 400 million people have diabetes and they are highly at risk of NAFLD and Carcinoma Therefore earlier prediction of Liver disorder can save people from Liver cancer and Cirrhosis. Implementation of Automated diagnostic system to the Liver disease dataset can reduce timing for prediction of disease and enhance accuracy. The primary goal of this work is to predict occurrence of Liver disease in provided dataset and their classification using machine learning algorithms.

3 Literature Review

Rahman et al. [12] (2019) proposed study on UPI machine learning Repository Dataset of AP. It utilised six machine learning techniques—LR, KNN, DT, SVM, NB, and RF—and performance was evaluated based on accuracy, precision, recall, and F-1 Score. Consequently, LR had the best accuracy (75%), whereas NB had the worst accuracy (53%). In terms of precision, LR scored the most, at 91%, and NB scored the lowest, at 36%. SVM scored the greatest when sensitivity was taken into account (88%), whereas KNN scored the lowest (76%). In term of F1-score LR got highest score 83% and NB got worst 53%. With respect to specificity DT obtained highest value 48% and LR got worst 47%.

Kalaviselvi and Santhoshmi (2019) [13] proposed a comparative study of various ML techniques to predict Liver Disease at early stage. The algorithms are examined and comparison of these algorithms performed on the basis of factors like accuracy, execution and precision to obtain best solution.

Kumari (2013) [14] proposed a hybrid approach to medical decision support system by this approach physicians can take pre-decision regarding Hepatitis disease. The dataset of hepatitis disease collected from UCI ML repository which contains 19 attributes. Techniques like SVM, Genetic algorithm, Clustering, Artificial Neural Networking and LR have used in this study. As a result, highest accuracy 96.77 has been obtained from LFDA-SVM and GA-RBEF for automatic Diagnosis of Hepatitis.

Nahar and Ara (2018) [15] used different types of Decision Tree techniques for early prediction of Liver disorder. The dataset of Liver disease was collected from UCI which contain 11 attributes for prediction of Liver disease. The decision three techniques used are J48, LMT, Random Forest, Random Tree, REPTree, Decision stamp and Hoeffding.

Ejiofor and Ugwu (2015) [19] proposed a hybrid system utilizing fuzzy logic system with SVM has been applied on the dataset of Cirrhosis and Hepatitis patient. As a result, they found better accuracy of this hybrid model for prediction of Liver Disorder.

Sultan (2012) [20] built a classification model in the form of a Tree structure to predict stages of Fibrosis in the patient with Chronic Hepatitis C genotype 4 in Egypt. 158 serum sample were collected from patient with Chronic Hepatitis C for analysis and 92.5% accuracy has obtained by Decision Tree.

Mitra and Samanta (2017) [24] extracted features using an RS-based feature selection technique. Laven-berg Marquardt algorithms and incremental back propagation learning networks (IBPLN) are employed as classifiers. Performance prediction uses CCR, sensitivity, specificity, and AUC. The dataset of UCI Hepatitis has been taken which contains 155 records with 20 attributes. As a result, the selected diagnosis of seven deduced features is: Age, Steroid, Billirubin, Alk phosphate, SGOT, Albumin, and Protein.

The Harsha Pakhale and Xaxa (2016) [26] surveyed many data mining techniques like DT, C4.5, CART, CHAID, Random Forest, SVM, Bayesian Net, MLP for classification of Liver patient. This paper concluded the summary of the data mining technique for prediction of Liver Disease.

Aravind et al. (2020) [30] used an advanced LIVERFASt to evaluate best accuracy of machine learning biomarker algorithms for diagnosis of Liver Fibrosis, Inflammation activity and Steatosis. The database of 13,068 patient records has been used which were collected from 16 sites of across Asia. They concluded modified SAF scores were generated by LIVERFASt provides an easiest and convenient diagnosis of NASH and NAFLD.

Vishwanath et al. (2010) [34] proposed Naïve Bayes algorithm for prediction of Liver disease in patient. The enzyme dataset was used in this study and the NB algorithm implemented for classification and prediction. During the literature

review, it was discovered that this algorithm is the most reliable and accurate. In order to make the procedure more seamless and efficient for the patient, the diagnosis result is delivered through email.

Kuppan and Manoharan (2017) [37] implemented J48 and Naïve Bayes to determine Liver disease using common variables such as alcohol consumption, smoking, obesity, diabetes, contaminated food intake, and a history of Liver disease. Sample data is collected from a number of hospitals that all have patients with liver disorders and are undergoing liver function testing. They concluded People suffer from liver disease at a higher rate than women. According to the results of the poll, people aged 35 to 65 are the most impacted by liver disease. People who drink heavily are more likely to get liver disease. People who smoke have a liver disorder, which affects 26% of the population. Obesity Patient People for Liver Disorder affects 22% of the population. 4% of persons are suffering with Liver Disorder.

Juliet et al. (2017) [38] presented a comprehensive assessment of various data mining-based strategies for the classification and prediction of Liver disease on USA Liver patient dataset.

Ramalingam et al. (2018) [39] proposed a survey of various classification approaches like logistic regression, SVM, random forest, KNN, decision tree, neural network, and ensemble method for disease prediction, notably in the case of liver disorders. Many existing strategies take into account the dataset that has been examined. The datasets they've looked at are connected to liver diseases like hepatitis and hepatocellular carcinoma.

Thangaraju and Mehala (2015) [40] implemented seven classification techniques as Bayes Net, Naïve Bayes, Multilayer perceptron, Decision Table, S.No., J48, and REPTree for identification of Liver cancer at earlier stage. The Breast Cancer dataset from UCI as taken for implementation of all these algorithms. They came to the conclusion that when there are more attributes, the accuracy can reach 100%, and that as the number of attributes decreases, the accuracy decreases.

Jangir et al. [41], Choubey et al. [42–48, 50, 52, 54, 65, 66] have used similar data science and machine learning algorithms for the identifications and predictions of medical diabetes. The idea conceived through the review [49, 53, 57–60, 63, 64] of many published articles, text and references like classification techniques diagnosis [51] for leukaemia, classification techniques diagnosis [55, 56] for heart disease, classification techniques diagnosis [61] for dengue, image detection [62] using computer vision are found to be of great help in accomplishment of the present work.

Table 1 consists the summary of existing works for liver disease. It consists the summarization of Dataset used, Techniques and Tools used, Purpose, Significance, Issues and Accuracy for the particular liver disease.

Table 2 is showing summary of Future Work over Existing work for the liver disease.

Table 1 Summary of existing works for liver disease

Ref. No.	Dataset	Techniques and tools used	Purpose	Significance	Issues	Accuracy
[1]	ILPD Dataset	Decision tree, Naïve-Bayes, K – nearest neighbor, WEKA, RapidMiner, Tanagra, Orange, KNIME	Classification algorithm have been compared by using these five data mining and KD tools.	Obtained KNIME tool with highest accuracy for classification algorithms	Performance of Naïve Bayes has found too lowest.	Decision tree: Tanagra (72.21%), Orange (65.18%), (KNIME) (86.58%)
[2]	Liver Disorder dataset	J48, Naïve Bayes, ANN, zeroR, IBK, VFI	Classification of liver diseases by using these classifiers and their comparison	Obtained better classifier for liver disease prediction	Performance of naïve bayes is not significant	J48 (68.97%), zeroR (57.97%), IBK (62.89%), NB (55.36%), VFI (60.28%), ANN (71.59%)
[3]	Two datasets have used one of BUPA and another of hepatitis	ANN	ANN algorithm has used to predict overlapping syndrome in liver disease	It will save time and effort of physicians	It requires more than 25 attributes for better accuracy	99.43%
[4]	Liver disease dataset	Random Forest, logistic regression, separation algorithm	Comparative analysis of various papers which are based on classification algorithms	Predicted 100% accuracy of random Forest for identification of liver disease	It can the model too slow ineffective	RF (100%)
[5]	ILPD dataset	DT, K-NN, MLP, NB, logistic, random F	Proposed RF and logistic methods and compared them with previously existed methods	Improved accuracy for diagnosis of liver disease	Used only liver patient dataset from India	LR (72.7%)

[6]	Ap liver disorder dataset and UCLA dataset	NB, KStar, Bagging, LMT, REP	Comparison of these classification algorithms in between two dataset	K* algorithm found to be most accurate algorithm for rapid identification of liver disease	Nb needs large number of records for better accuracy	NBs (39%), Kstar (100%), bagging (88%), REP (79%), LMT (75.4%)
[7]	Liver disease dataset	SVM, NB, C4.5 decision tree	Predicted liver disease by help of these techniques	Compared these techniques to obtain best model	Needs more number of algorithms to compare	
[8]	Liver disease dataset	ANFIS FCM	Proposed hybrid technique of ANFIS and FCM for diagnosis of liver disease	Hybrid model had higher accuracy as compare to simple model	Mean absolute error appeared higher as compare to few classifiers	
[9]	NAFLD disorder dataset	KNN, SVM, LR, NB, BN, C4.5, AdaBoost, Bagging, RF, HNB, AODE, FLI, HIS	Performed a cross-sectional study to find screening and predictive model for NAFLD by using ML algorithms	LR obtained highest accuracy for prediction of NAFLD liver disease.	The result obtained from decision model is dependent on k chosen value	LR (83.41%)
[10]	Liver disease dataset from Tamil Nadu	Fuzzy K-means classifier	Categorised liver disorders by using feature selection and fuzzy K-means classifier	Fuzzy model has been classified liver disease with more efforts.	Fuzzy classification gives approximation rather than exact	94%

(continued)

Table 1 (continued)

Ref. No.	Dataset	Techniques and tools used	Purpose	Significance	Issues	Accuracy
[11]	Chronic viral hepatitis C dataset	FAHP ANFIS	Proposed application of FAHP and ANFIS in the critical medical problem of fibrosis diagnosis	Obtained highest accuracy for prediction of fibrosis disease.	Number of epoch arose slow processing of model	93%
[16]	Dataset of blood test result from India	KKN, logistic regression, decision tree, SVM, RF	Used these techniques for prediction and identification of liver disease	By the help of patient record physicians can determine whether the patient has disease or not	Required more number of records for better performance.	LR (71.42%), KNN (64%), SVM (70.28%), DT (64.57%), RF (66.85%)
[17]	ILPD	ANN	Classification of liver disease by the help of ANN algorithm.	Reduced system complexity.	Consumed great extent of time.	95.40%
[18]		K-means clustering, C4.5 DT	These techniques have applied on the online dataset for prediction of chronic liver disease	C4.5 achieved highest accuracy as compare to K-means which may useful in earlier diagnosis.	Overfitting problem has arisen.	K-means (93.47%), C4.5 (94.36%)
[21]	Liver patient dataset from UCI	NB, LR, random Forest, XG boost, ANN	Predicted liver disease by using these techniques and compared their accuracy for better performance	ANN can be considered as best model for diagnosis of liver disease with highest accuracy.	Need to increase number of epoch to obtain better accuracy	ANN (100%), LR (66%), NB (52%), RF (66%), XG boost (88%)

[22]	ILPD	LR, SVM, RF, AdaBoost, bagging	These ML algorithms were applied on the ILPD to predict liver disease by the enzyme content at an early stage	It can be applied in healthcare center for earlier prediction of liver disease	Required more dataset for highest accuracy	LR (73.5%), SVM (70.94%), RF (66.66%), AdaBoost (74.35%), Bagging (72.64%)
[23]	Dataset of respiratory disease	FES	Fuzzy expert system was developed for diagnosis of cystic fibrosis	It will increase diagnosis rate of liver disease and reduce cost of the test	Complex to handle.	92.86%
[25]	One ILPD from UCI and another from pathological lab	k-means, AGNES, DBSCAN, OPTICS, EM	These techniques are applied on ILPD dataset and their performances have measured			k-means (64.28%), AGNES (61.32%), DBSCAN (51.39%), OPTICS (54.67%), EM (59.65%)
[27]	Liver disease dataset from Andhra Pradesh state of India	NB, J48, random Forest, k-star	Classification of liver disease was performed by the help of these techniques	It was implemented to reducing time of diagnosis		NB (60.6%), k-star (67.2%), J48 (71.2%), random Forest (74.2%)
[28]	Hepatitis C dataset from UCI	DT, NB, LR, SVM	These techniques were applied on hepatitis C dataset to predict whether patient dies or live.			SVM (76.92%), NB (69.23%), DT (82.05%), LR (87.17%)

(continued)

Table 1 (continued)

Ref. No.	Dataset	Techniques and tools used	Purpose	Significance	Issues	Accuracy
[29]	Hepatitis dataset from UCI	BVM, C4.5, C-RT, CS-CRT, CS-MCA, C-SVC, CVM, ID3, KNN, LDA, MLP, NBC, PLS-LDA, Rnd tree	Three feature selection and these classification algorithms were applied on the hepatitis dataset.	Improved accuracy	Feature selections have found not suitable for this dataset	BVM, CVM and Rnd tree obtained (100%) accuracy.
[31]	Liver dataset from Anrutha group of hospital media	NB, C4.5, AD tree, SVM, RBF, MLFFDNN	MLFFDNN has applied on liver disease to get better accuracy	Showed better accuracy than earlier existing algorithms for prediction of liver disease	Overlapping problem and took a long time to complete the work	NB (71%), C4.5 (97%), AD tree (92%), SVM (75%), RBF (83%), MLFFDNN (98%).
[32]	Indian liver patient Dataset from UCI	KNN, SVM, random Forest, k-means, hybrid approach, Jupiter notebook	A hybrid algorithm has proposed which is based on clustering and classification techniques for prediction of liver disease	Proposed hybrid method can detect liver disease at earlier stage with high accuracy.	New algorithms can be proposed to get more accuracy than proposed model.	KNN (72.05%), LR (73.23%), SVM (71.12%), RF (92.06%), hybrid approach (95%).
[33]	ILPD from UCI	SVM, Kernal function	Classification of liver disease has performed by SVM and its performance evaluated by different kernel function	Diagnosis of patient can be done by more effectively	Linear Kernel has obtained highest accuracy when compared to others	Linear (81.67%), quadratic (81.16%), MLP (81.16%), RBF (82.53%), polynomial (81.67%)
[35]	Liver disorder Dataset from UCI	NB, FT tree, K-star	Predicted liver disease by the help of these three techniques.	FT tree can be used to effective diagnosis of liver disease	As the number of attributes decreases performance also decreases.	NB (96.52%), FT tree (97.10%), k-star (83.47%)
[36]	LFT dataset	SVM, DT, linear discriminant, logistic regression	These techniques have applied on this dataset for liver disease prediction	It will helpful in earlier diagnosis	Poor performance with non-linear data and high time requirement	SVM (82.7%), DT (94.9%), LR (95.8%).

Table 2 Summary of the future works over the existing works

Authors with Ref. No.	Existing work	Future work
[7]	Different data mining classification algorithms were used to obtain better result and compared with the earlier existed prediction methods.	Hybrid approach can use to obtain better performance accuracy for prediction of liver disease.
[9]	Eleven machine learning techniques were used to find screening and prediction model, compared them with each-other to obtain higher value for accuracy, specificity, precision, recall.	This study can be improved by using these classification algorithms in biopsy result for verification of the prediction power of machine learning model.
[11]	CDSS was developed by using two soft computing techniques FAHP and ANFIS for diagnosis of fibrosis, fibrosis stage calculated by a mathematical model.	This work can extend by applying these techniques to other medical problems
[13]	A comparative study of various ML and classification algorithms to predict probability of liver disease.	We can compare decision tree, ANFIS and KNN in term of accuracy, sensitivity, specificity and precision.
[16]	Classification algorithms KNN, LR, SVM, DT were used for identification and prediction of liver disease.	IOT application need to build up for prediction of blood test report by UI manually to get whether the patient has liver disease or not.
[17]	Classification of liver disease performed by ANN algorithm and ANN obtained highest accuracy as compare to existed methods.	Liver disease characterization could be improved by utilizing advancement system and diagnosis the various types of liver sicknesses.
[18]	Predicted chronic liver disease based on the impact of life quality attributes by the help of k-means and c4.5. c4.5 achieved higher accuracy.	Diagnosis of multiple disease will be performed on the basis of life quality attributes.
[31]	Detection of liver disease has performed by using MLFFDNN (multi-layer feed forward deep neural network). It obtained 98% accuracy as compare to conventional techniques.	Boosting technique will be used for improving accuracy and dealing with imbalanced dataset.
[34]	Predicted liver disease in patients by using Naïve byes algorithm and they concluded that NB is most effective in term of reliability and accuracy	We can enhance accuracy by applying multiple other classification techniques on this system and another future work is application of NB on other disease like diabetes and heart disease
[36]	A model for liver disease prediction has developed by using MATLAB2016, SVM, LR and DT. LR achieved highest accuracy among them	After a patient is diagnosed with a liver problem, more study should be done to look at the tumour features of the patient. Large datasets can also be utilised for training the model and determining methods.

4 Materials and Methods

The work has been illustrated into two parts: (a) Dataset and (b) Proposed Algorithms.

4.1 Dataset

The dataset of ILPD has been obtained from the UCI Machine Learning repository. The dataset contains 416 liver patient records, 167 non-liver patient records and 11 attributes collected from North East of Andhra Pradesh. In this data collection, there are 142 female patient records and 441 male patient records. Total 583 records are present in this dataset. Table 3 contains the details of attributes.

4.2 Algorithms Description

In this work, we used Jupyter Notebook to determine the prediction of Liver disease using Python. Here we have presented workflow diagram of our implemented algorithms for downloaded dataset. After implementation of algorithms, we compared their performances based on their accuracy, precision, recall, f1-score and support values. The algorithms employed in the model are described in depth further down. Figure 1 depicts the proposed architecture:

Table 3 Attributes of ILPD dataset

No	Attributes
1	Age
2	Gender
3	Total bilirubin
4	Direct bilirubin
5	Alkaline phosphatase
6	Alamine aminotransferase
7	Aspartate aminotransferase
8	Total protein
9	Albumin
10	Albumin and globulin ratio
11	Dataset

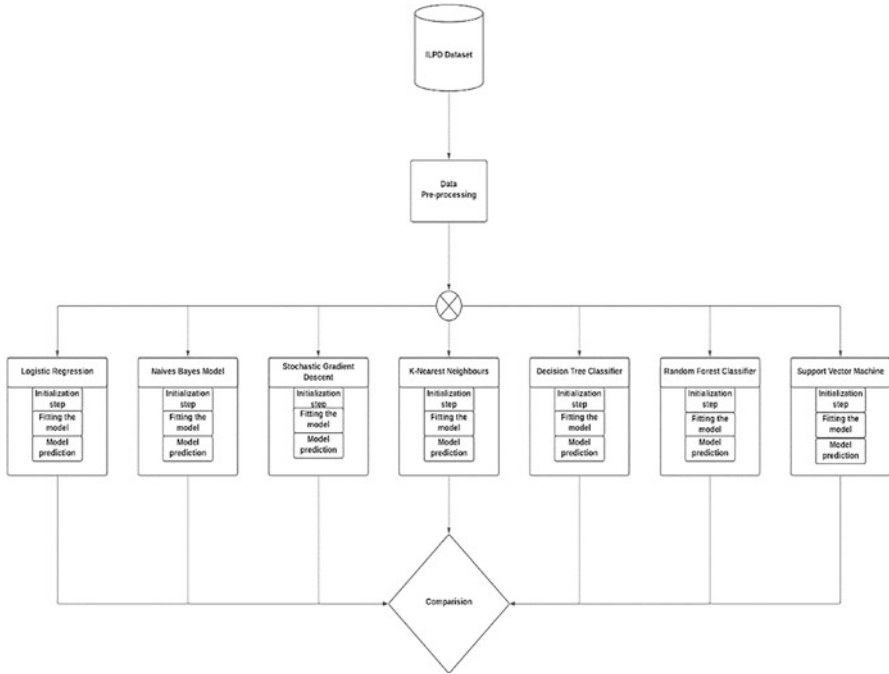


Fig. 1 Proposed architecture

4.2.1 Logistic Regression

Logistic regression (LR) is a supervised classification algorithm which is being used in prediction of the probability of a target variable. LR uses binary dependent variable and provides result in the form of two possible output. Setting of threshold value is extremely necessary aspect for LR to become a classification technique. LR is easier in implementation, interpretation, and very efficient to train a model. LR can be binomial, multinomial and ordinal, binomial target variable can have only two possible outputs where multinomial target variable can have three or more unordered possible output and ordinal deals with ordered categories. Linear regression equation is very similar to LR equation. In this model input values (x) are combined linearly using weights or coefficient for prediction of an output value (y).

The Fig. 2. illustrates the schematic of a LR classifier which are mentioned below:

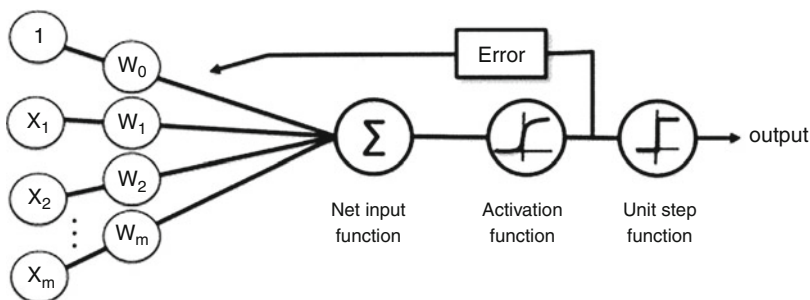


Fig. 2 Schematic of a logistic regression classifier

The algorithm of logistic regression is noted below:

Algorithm 1. Logistic Regression

Input: X_{train} , y_{train}

1. Proposed Logistic Regression
2. `model01= LogisticRegression()`
- # Fitting a model
3. `model01.fit(X_train,y_train)`
- # Model Prediction
4. `model01.predict(X_test)`
- # Summary of the model
5. `model01.score(X_test, y_test)`
6. End Procedure

Output: accuracy, precision, recall, f1-score, support, roc

4.2.2 Gaussian Naïve Bayes

Gaussian Naïve Bayes (GNB) is a mutant of NB that obeys Gaussian normal distribution. GNB accepts characteristics with continuous values and models them all as fitting a Gaussian normal distribution. Assume the data has a Gaussian distribution with no parameter covariance in order to construct a straightforward model. To fit this model, all that is needed to characterise this type of distribution is to compute the mean and standard deviation of the data within each label. Assume that the data is distributed in a Gaussian fashion with no parameter covariance in order to construct a straightforward model.

Figure 3 illustrates the Gaussian Distribution which are mentioned below:

The algorithm of Gaussian Naïve Bayes is noted below:

Algorithm 2. Gaussian Naïve Bayes

Input: X_{train} , y_{train}

1. Proposed Gaussian Naïve Bayes
2. `model02= GaussianNB()`
- # Fitting a model
3. `model02.fit(X_train,y_train)`
- # Model Prediction

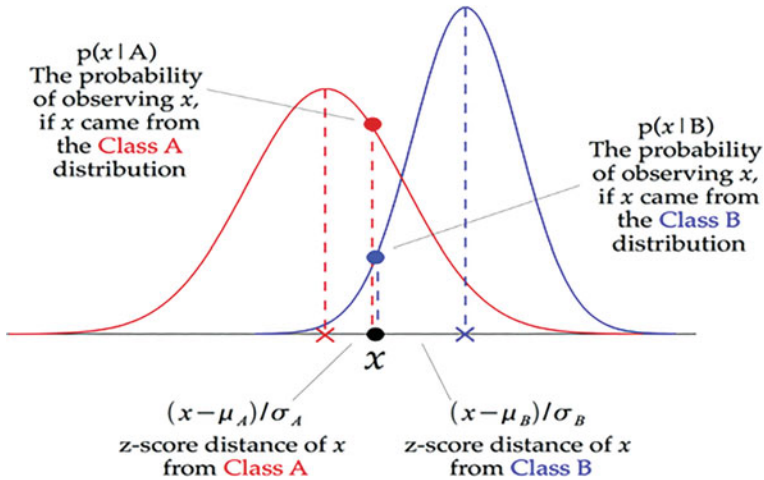


Fig. 3 Gaussian Distribution

```

4. model02. predict(X_test)
# Summary of the model
5. model02. score(X_test, y_test)
6. End Procedure
    
```

Output: accuracy, precision, recall, f1-score, support, roc

4.2.3 Stochastic Gradient Descent

In machine learning and deep learning, gradient descent is a well-known optimization technique that can be used to almost all learning algorithms. The gradient of a function is the slope. It establishes how much one variable will vary in response to adjustments made to another. A system or process that is subject to chance is referred to as “stochastic.” As a result, in Stochastic Gradient Descent only a few samples are randomly chosen for each iteration as opposed to the complete data set. SGD (Stochastic Gradient Descent) is a straightforward but effective optimization algorithm for determining the values of coefficients of functions that minimise a cost function.

Figure 4 shows the schematic diagram of Gradient descent.

The algorithm of Stochastic Gradient Descent is noted below:

```

Algorithm 3. Stochastic Gradient Descent
Input: X_train, y_train
1. Proposed Stochastic Gradient Descent
2. model03= SGDClassifier()
# Fitting a model
3. model03. fit(X_train,y_train)
# Model Prediction
4. model03. predict(X_test)
    
```

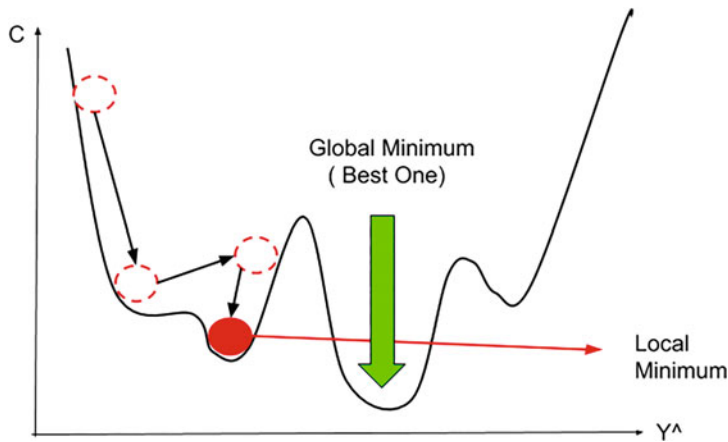



Fig. 4 Stochastic Gradient Descent

```
# Summary of the model
5.model03. score(X_test, y_test)
6. End Procedure
```

Output: accuracy, precision, recall, f1-score, support, roc

4.2.4 K-Nearest Neighbours

One of the simplest Machine Learning algorithms is K-Nearest Neighbor, which is based on the Supervised Learning method. Due to its non-parametric nature, which implies that no underlying assumptions about data distribution are made, it is frequently employed in real-world applications. KNN, is a data categorization method that calculates how probable a data point is to belong to one of two groups based on which group the data points closest to it belong to. KNN is also referred to as a lazy learner algorithm since it saves the training set and performs a classification operation on it rather than learning from it immediately. The KNN method simply stores the information during the training phase and classifies incoming data into a category that closely resembles the training data.

Figure 5 shows KNN classifier.

The algorithm of K-Nearest Neighbours is noted below:

```
Algorithm 4. K-Nearest Neighbours
Input: X_train, y_train1. n_neighbors = 7
2. Proposed K-Nearest Neighbours
3. model04= KNeighborsClassifier(n_neighbors = 7)
# Fitting a model
4. model04. fit(X_train,y_train)
# Model Prediction
5. model04. predict(X_test)
# Summary of the model
```

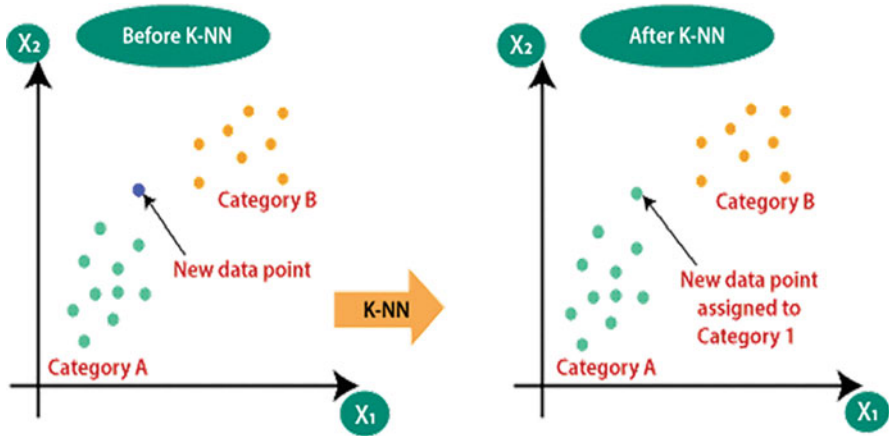


Fig. 5 KNN classifier

```
6.model04. score(X_test, y_test)
7. End Procedure
```

Output: accuracy, precision, recall, f1-score, support, roc

4.2.5 Decision Tree Classifier

Decision Trees are made up of nodes, branches, and leaves. A characteristic (or feature) is represented by each node, a rule (or option) by each branch, and a consequence by each leaf. A Tree’s depth is determined by the number of levels it has, omitting the root node. DTs approach data from the top down, trying to group and classify similar observations among them and looking for the best criteria to divide the dissimilar observations until they reach a certain level of similarity. In order to determine the appropriate partitioning, the splitting can be binary (which divides each node into a maximum of two subgroups) or multiway (which divides each node into a maximum of four subgroups) (which splits each node into multiple sub-groups, using as many partitions as existing distinct values).

Nodes, branches, and leaves make up DTs. Each leaf denotes an outcome, each branch denotes a rule (or choice), and each node denotes an attribute. The number of levels, excluding the root node, determines the depth of a tree. Figure 6. Illustrates the decision tree of two levels.

The algorithm of decision tree classifier is noted below:

```
Algorithm 5. Decision Tree Classifier
Input: X_train, y_train1. max_depth = None , random_state = 1
2. max_features = None, min_samples_leaf =20
3. Proposed Decision Tree Classifier
4. model05= DecisionTreeClassifier(max_depth = None ,
random_state = 1 , max_features = None,
```

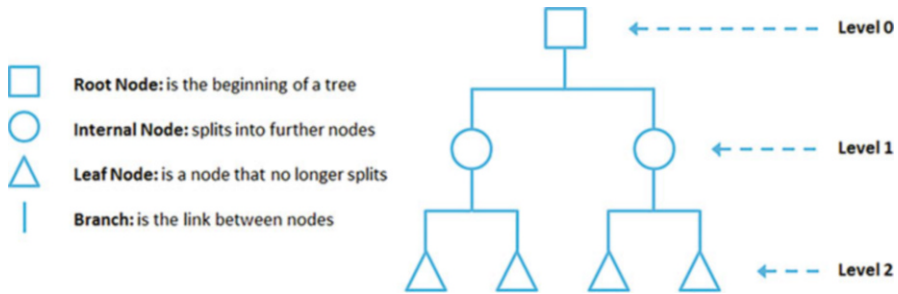


Fig. 6 Decision Tree of two levels

```

min_samples_leaf =20)
# Fitting a model
5. model05. fit(X_train,y_train)
# Model Prediction
6. model05. predict(X_test)
# Summary of the model
7. model05. score(X_test, y_test)
8. End Procedure
  
```

Output: accuracy, precision, recall, f1-score, support, roc

4.2.6 Random Forest

A Random Forest is an ensemble method for solving classification and regression problems that combines many decision trees with a method known as Bootstrap and Aggregation, also referred to as bagging. The fundamental idea is to combine several decision trees to determine the outcome rather than relying just on individual decision trees. Because the random forest model is made up of so many largely uncorrelated models (trees), it performs better as a whole than any of its individual component models. The reason for this astounding result is that the trees protect one another from one another's errors, provided that no two of them commit the same error. Many trees will be right while some may be wrong, allowing the group of trees to move in the right direction. Figure 7 illustrates tree of Random Forest.

The algorithm of Random Forest is noted below:

Algorithm 6. Random Forest

```

Input: X_train, y_train1. n_estimators=80, max_depth=8,
random_state=0
2. cv=5
3. Proposed Random Forest
4. model06= RandomForestClassifier(n_estimators=80, max_depth=8,
random_state=0)
# Fitting a model
5. model06. Fit(X_train,y_train)
# Model Prediction
6. model06. Predict(X_test)
  
```

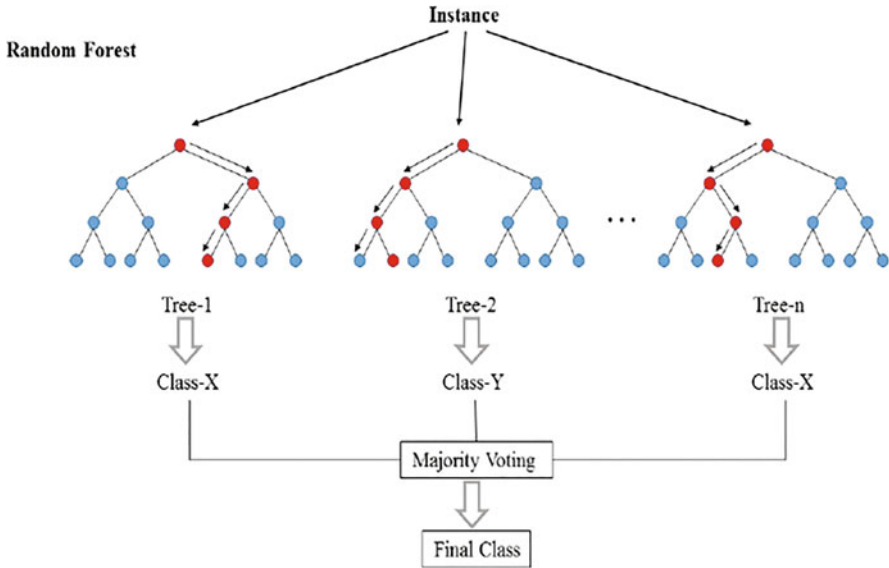


Fig. 7 Random forest

```
# Summary of the model
7. model06. Score(X_test, y_test)
8. End Procedure
Output: accuracy, precision, recall, f1-score, support, roc
```

4.2.7 Support Vector Machine

A popular supervised learning approach is Support Vector Machine (SVM). In order to categorise n-dimensional space into classes and make it simple to add additional data points in the future, the SVM algorithm looks for the best line or decision boundary. The optimal choice boundary is known as a hyperplane. SVM is used to choose the extreme points and vectors that help build the hyperplane. The approach is referred described as a Support Vector Machine since support vectors are the extreme situations. A kernel is used by the SVM method to convert an input data space into the desired format. A method employed by SVM to convert a low-dimensional input space into a higher-dimensional space is known as the kernel trick.

Figure 8 shows the basic SVM model.

The algorithm of SVM is noted below:

```
Algorithm 7. Support Vector machine Model
Input: X_train, y_train1. C=0.025, random_state = 0 , gamma=0.01
2. kernel= "linear", cv=5
3. Proposed Support Vector machine Model
```

Fig. 8 Support Vector machine Model

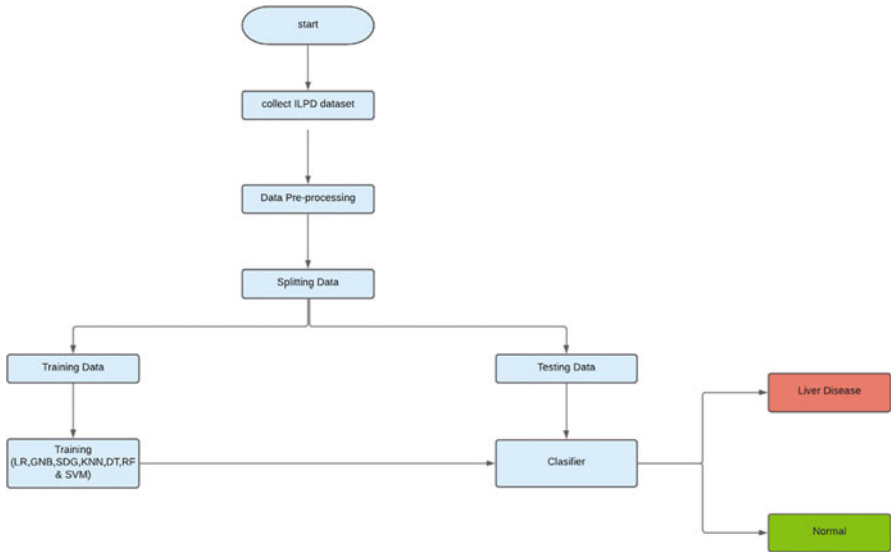
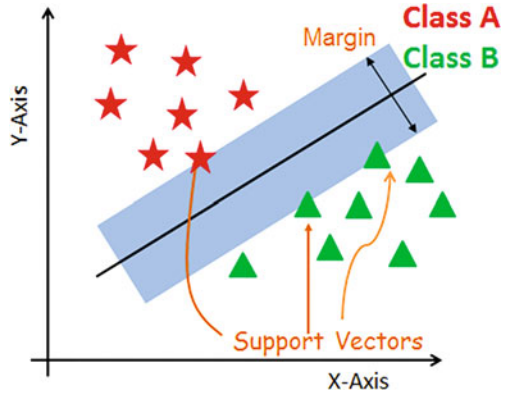


Fig. 9 Dataflow diagram

```
4. model07= SVC(kernel= "linear",C=0.025, random_state = 0 ,
gamma=0.01)
# Fitting a model
5. model07. Fit(X_train,y_train)
# Model Prediction
6. model07. Predict(X_test)
# Summary of the model
7. model07. Score(X_test, y_test)
8. End Procedure
```

Output: accuracy, precision, recall, f1-score, support, roc

Figure 9 shows the dataflow diagram of this study. We have downloaded the ILPD dataset from UCI Repository and performed feature extraction process, after

doing feature extraction dataset have been pre-processed using Python and split into training and testing. Thereafter we implemented ML algorithms and prediction of disease has been performed.

5 Experimental Results and Discussion

Table 4 shows the results of implemented algorithms:

The generated confusion matrix of LR and GNB are shown in below Figs. 10 and 11:

Table 4 Results of implemented algorithms

Parameters	LR	GNB	SGD	KNN	DT	RF	SVM
Accuracy	0.712	0.565	0.717	0.662	0.751	0.696	0.717
Precision	0.717	0.887	0.717	0.731	0.798	0.759	0.717
Recall	1.000	0.452	1.000	0.837	0.875	0.846	1.000
F1-score	0.835	0.599	0.835	0.780	0.835	0.800	0.835
Support	104	104	104	104	104	104	104
ROC	0.500	0.653	0.500	0.528	0.657	0.582	0.500

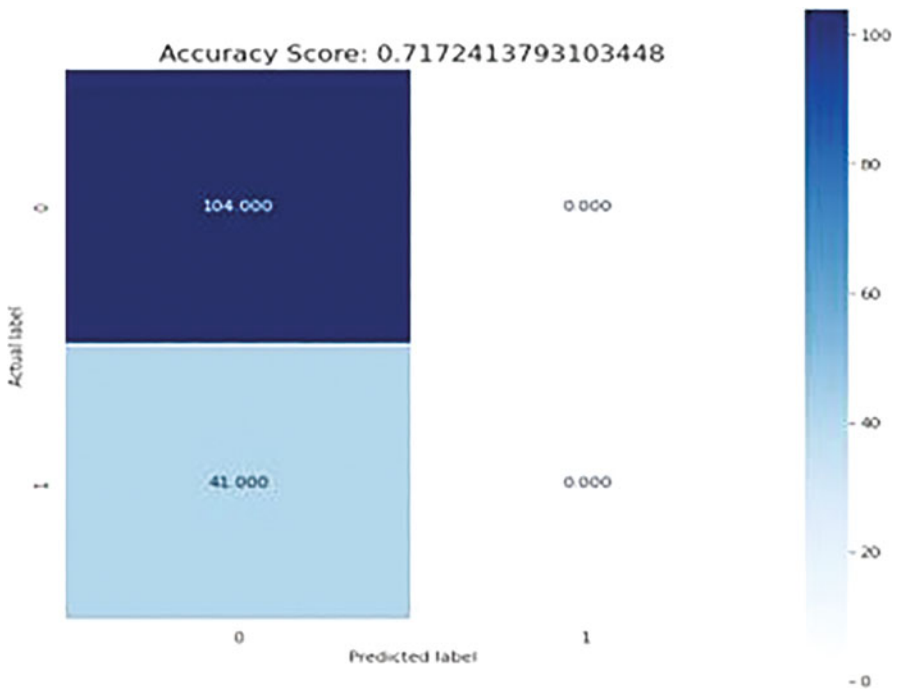


Fig. 10 Confusion matrix of logistic regression

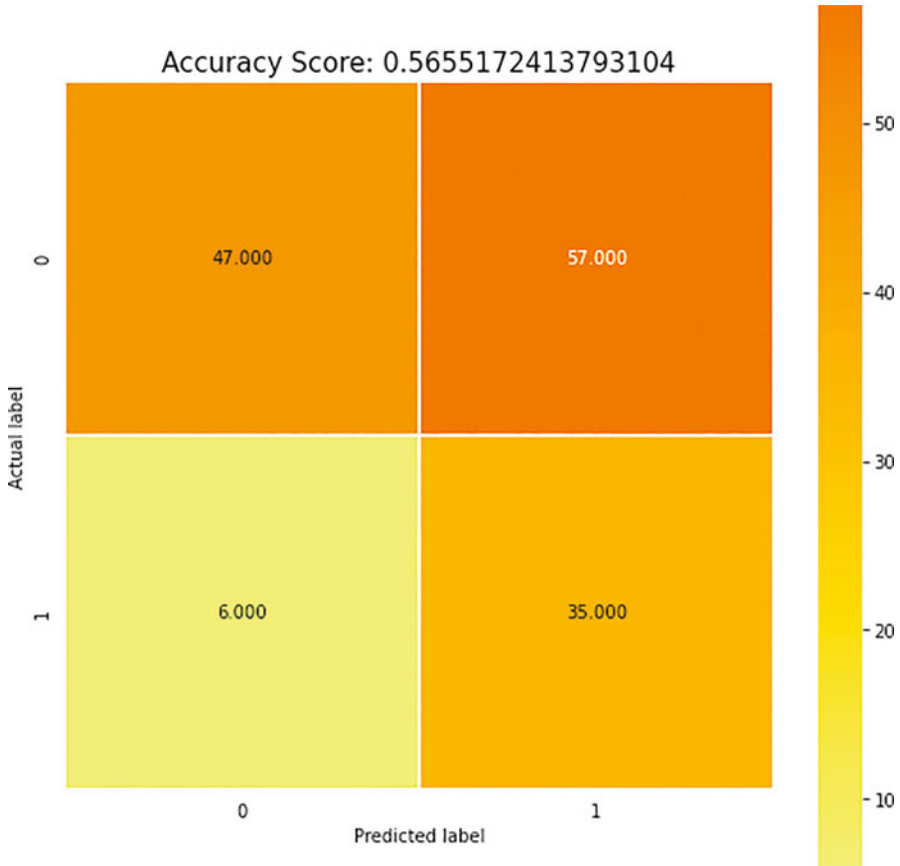


Fig. 11 Confusion matrix of Gaussian Naïve Bayes

The generated confusion matrix of Stochastic Gradient Descent and K-Nearest Neighbour are shown in below Figs. 12 and 13.

The generated confusion matrix of Decision Tree and Random Forest are shown in below Figs. 14 and 15.

The generated confusion matrix of SVM is shown in below Fig. 16.

Table 5 illustrates the comparison of proposed algorithms with existing algorithms.

6 Conclusion and Future Scope

Even a doctor will find it difficult to diagnose a liver ailment based on raw data, which is why many healthcare organisations are turning to machine learning

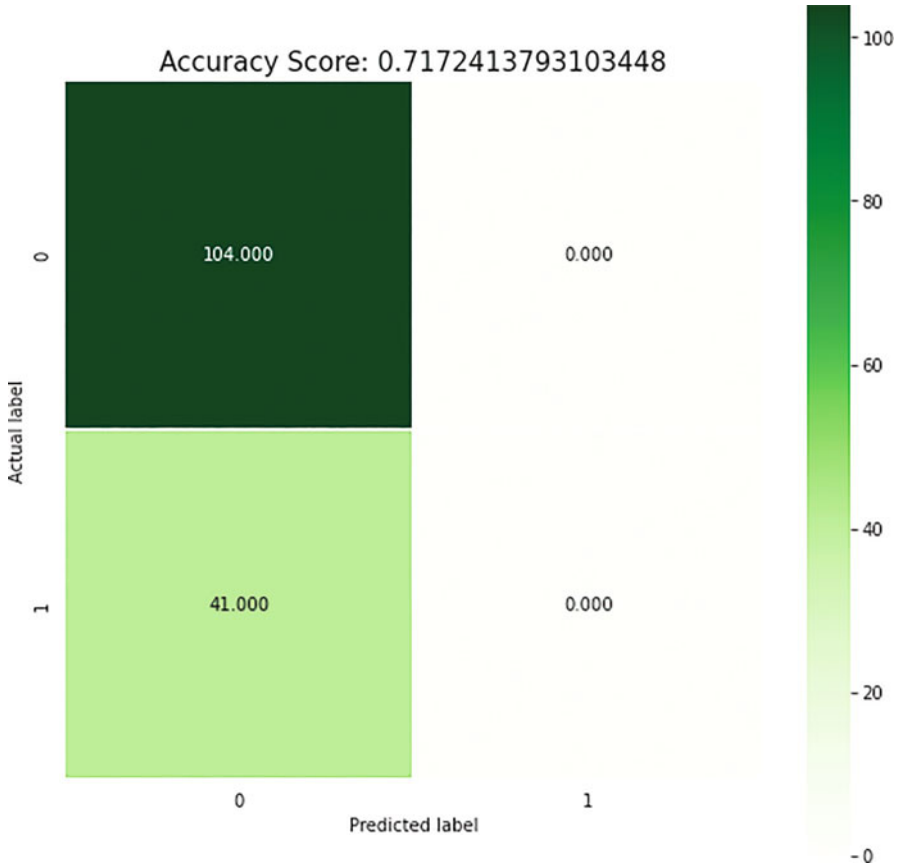


Fig. 12 Confusion matrix of Stochastic gradient descent

techniques to do it. In our Project we extract ILPD dataset from UCI Repository and Pre-processing was used to eliminate data with missing values, and Machine Learning models as Logistic Regression, Gaussian Naïve Bayes, Stochastic Gradient Descent, K-Nearest Neighbour, Decision Tree, Random Forest and Support Vector Machine were used. The performance of these algorithms was analysed on the basis of their accuracy where Decision Tree achieved the highest accuracy of 75.1%, followed by Stochastic Gradient Descent 71.7%, Support Vector Machine 71.7%, Logistic Regression 71.2%, Random Forest 69.6%, K-Nearest Neighbour 66.2%, and Gaussian Naïve Bayes obtained a worst accuracy with 56.5%. After implementation of all algorithms, we compared their performance with existing works and concluded that DT have achieved highest accuracy among them.

In Future the accuracy of classification will be enhanced by feature extraction and the big dataset can also be examined for training the model and determining algorithms. Another future work is application of these proposed algorithms on other diseases.

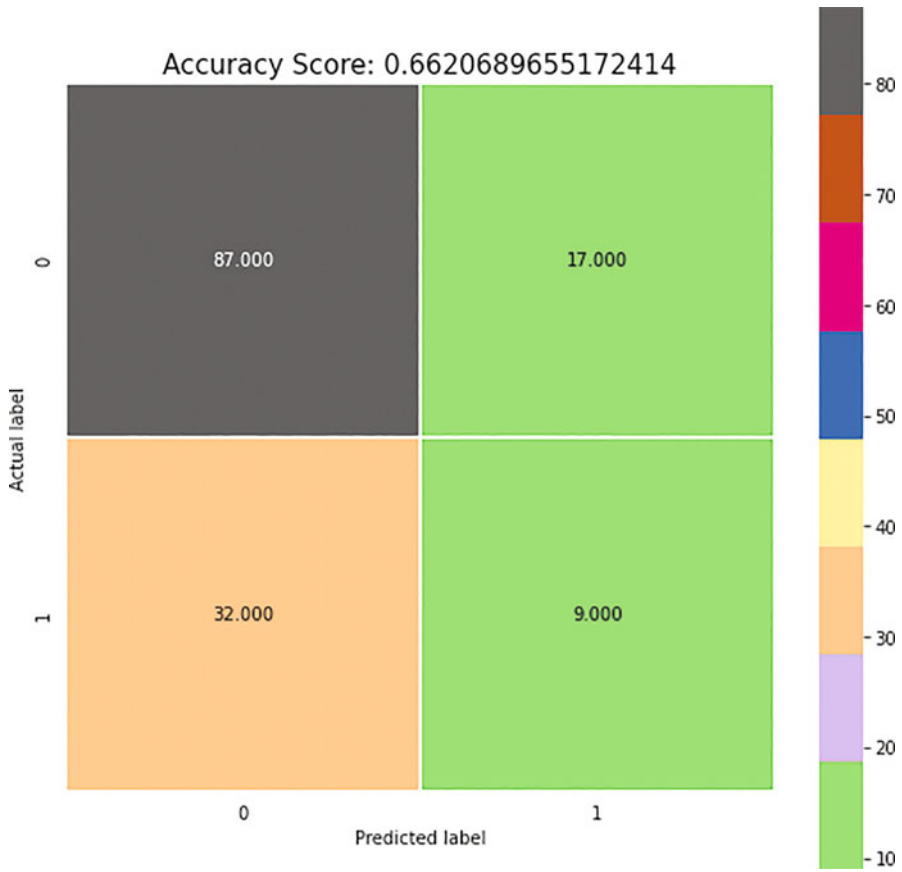


Fig. 13 Confusion matrix of K-Nearest neighbour

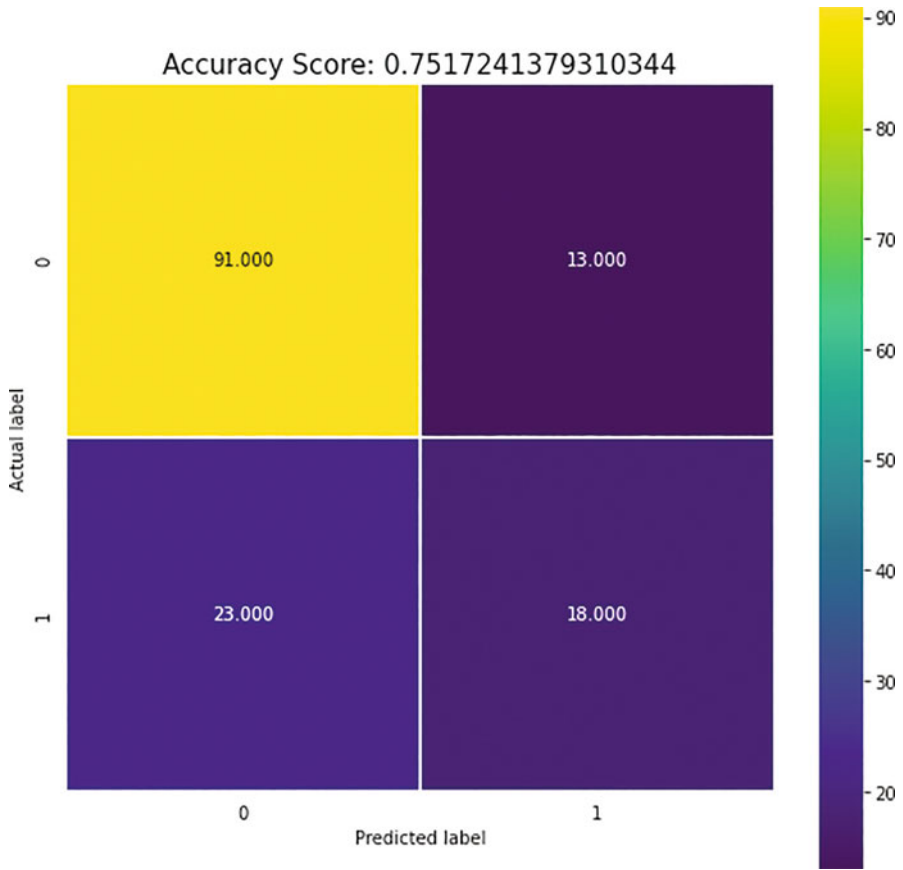


Fig. 14 Confusion matrix of decision tree

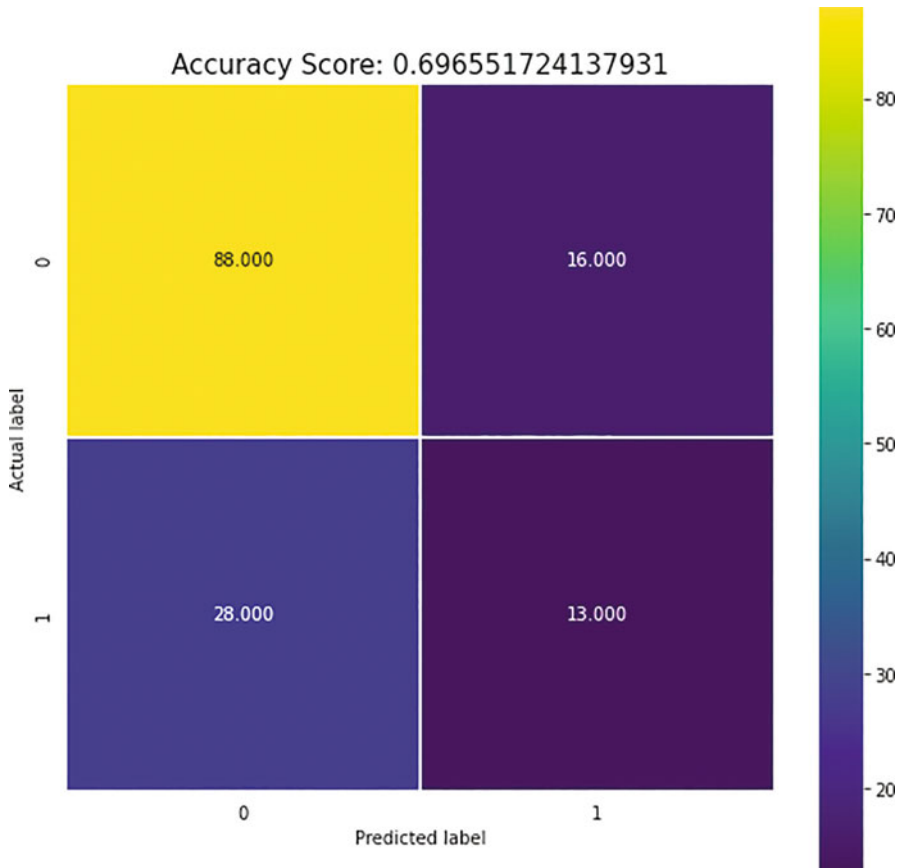


Fig. 15 Confusion matrix of random forest

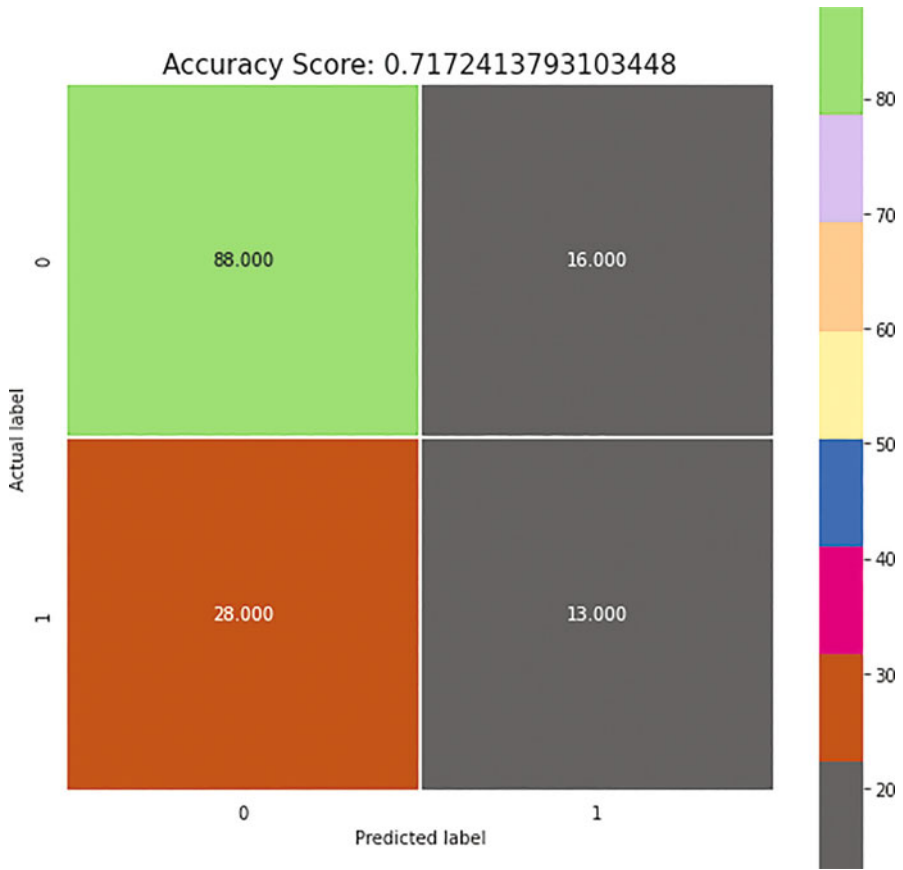


Fig. 16 Confusion matrix of SVM

Table 5 Comparison of proposed algorithms with existing algorithms

Sources	Algorithm	Accuracy
[25]	K-means	64.32%
	AGNES	61.32%
	DBSCAN	51.39%
	OPTIC	54.67%
	EM	59.65%
[27]	Naïve Bayes	60.6%
	Random Forest	67.2%
	K-star	71.2%
[16]	K-nearest neighbour	64%
	Logistic regression	71.42%
	Decision tree	64.57%
	SVM	70.28%
	Random Forest	66.85%
Our study	Decision tree classifier	75.1%
	Gaussian Naïve Bayes	56.5%
	Stochastic gradient descent	71.7%
	K-nearest neighbours	66.2%
	Logistic regression	71.2%
	Random Forest	69.6%
	Support vector machine	71.7%

References

1. Naik, A., & Samant, L. (2016). Correlation review of classification algorithm using data mining tool: WEKA, Rapidminer, Tanagra, Orange and Knime. *Procedia Computer Science*, 85, 662–668.
2. Baitharu, T. R., & Pani, S. K. (2016). Analysis of data mining techniques for healthcare decision support system using liver disorder dataset. *Procedia Computer Science*, 85, 862–870.
3. Alghamdi, M. A., Bhirud, S. G., & Alam, A. M. (2015). Physician's decision process for disease diagnosis of overlapping syndrome in liver disease using soft computing model. *International Journal of Soft Computing and Engineering (IJSCE)*, 4(6), 28–33.
4. Khan, B., Shukla, P. K., Ahirwar, M. K., & Mishra, M. (2021). Strategic analysis in prediction of liver disease using different classification algorithms. In *Handbook of Research on Disease Prediction through Data Analytics and Machine Learning* (pp. 437–449). IGI Global.
5. Jin, H., Kim, S., & Kim, J. (2014). Decision factors on effective liver patient data prediction. *International Journal of Bio-Science and Bio-Technology*, 6(4), 167–178.
6. Ghosh, S. R., & Waheed, S. (2017). Analysis of classification algorithms for liver disease diagnosis. *Journal of Science Technology and Environment Informatics*, 5(1), 360–370.
7. Kefelegn, S., & Kamat, P. (2018). Prediction and analysis of liver disorder diseases by using data mining technique: Survey. *International Journal of Pure and Applied Mathematics*, 118(9), 765–770.
8. Kour, H., Sharma, A., Manhas, J., & Sharma, V. (2018). Automated intelligent diagnostic liver disorders based on adaptive neuro fuzzy inference system and fuzzy C-means techniques. *Mody University International Journal of Computing and Engineering Research*, 2(2), 86–91.

9. Ma, H., Xu, C. F., Shen, Z., Yu, C. H., & Li, Y. M. (2018). Application of machine learning techniques for clinical predictive modeling: A cross-sectional study on nonalcoholic fatty liver disease in China. *BioMed Research International*, 3, 4304376.
10. Aneeshkumar, A. S., & Venkateswaran, C. J. (2015). A novel approach for liver disorder classification using data mining techniques. *Engineering and Scientific International Journal (ESIJ)*, 2, 2394–7179.
11. El-Sappagh, S., Ali, F., Ali, A., Hendawi, A., Badria, F. A., & Suh, D. Y. (2018). Clinical decision support system for liver fibrosis prediction in hepatitis patients: A case comparison of two soft computing techniques. *IEEE Access*, 6, 52911–52929.
12. Rahman, A. S., Shamrat, F. J. M., Tasnim, Z., Roy, J., & Hossain, S. A. (2019). A comparative study on liver disease prediction using supervised machine learning algorithms. *International Journal of Scientific & Technology Research*, 8(11), 419–422.
13. Kalaviselvi, R., & Santhoshmi, G. (2019). A comparative study on predicting the probability of liver disease. *International Journal of Engineering Research and Technology*, 8(10), 560–564.
14. Kumari, S. (2013). Computational intelligence in the hepatitis diagnosis: A review. *International Journal of Computer Science and Technology*, 4(2), 265–270.
15. Nahar, N., & Ara, F. (2018). Liver disease prediction by using different decision tree techniques. *International Journal of Data Mining & Knowledge Management Process*, 8(2), 01–09.
16. Varshney, N., & Sharma, A. (2020). Identification and prediction of liver disease using logistic regression. *European Journal of Molecular & Clinical Medicine*, 7(4), 106–110.
17. Anand, L., & Neelanarayanan, V. (2019). Liver disease classification using deep learning algorithm. *BEIESP*, 8(12), 5105–5111.
18. Sivakumar, D., Varchagall, M., Ambika, L. G., & Usha, S. (2019). Chronic liver disease prediction analysis based on the impact of life quality attributes. *International Journal of Recent Technology and Engineering (IJRTE)*, 7(6S5), 2111–2117.
19. Ejiofor, C., & Ugwu, C. (2015). Application of support vector machine and fuzzy logic for detecting and identifying liver disorder in patients. *IOSR Journal of Computer Engineering*, 17(3), 50–53.
20. Sultan, T. I., Khedr, A., & Sabry, S. (2012). Biochemical markers of fibrosis for chronic liver disease: Data mining-based approach. *International Journal of Engineering Research and Development*, 2(2), 08–15.
21. Priyanka.G, Mithun.S, Midhun Sakravarthy.V, Kishore.P. (2020). Machine learning based analysis for liver disorder prediction. *International Journal of Advanced Science and Technology*, 29(3), 9859–9867.
22. Idris, K., & Bhoite, S. (2019). Application of machine learning for prediction of liver disease. *International Journal of Computer Application Technology and Research*, 8(9), 394–396.
23. Hassanzad, M., Orooji, A., Valinejadi, A., & Velayati, A. (2017). A fuzzy rule-based expert system for diagnosing cystic fibrosis. *Electronic Physician*, 9(12), 5974.
24. Mitra, M., & Samanta, R. K. (2017). A study on UCI hepatitis disease dataset using soft computing. *Modelling, Measurement and Control C*, 78(4), 467–477.
25. Babu, K. S. P. M. P. (2017). A critical study on cluster analysis methods to extract liver disease patterns in Indian liver patient data. *International Journal of Computational Intelligence Research*, 13(10), 2379–2390.
26. Pakhale, H., & Xaxa, D. K. (2016). A survey on diagnosis of liver disease classification. *International Journal of Engineering and Techniques*, 2(3), 132–138.
27. Muthuselvan, S., Rajapraksh, S., Somasundaram, K., & Karthik, K. (2018). Classification of liver patient dataset using machine learning algorithms. *International Journal of Engineering and Technology*, 7(3.34), 323.
28. Bhargav, K. S., Thota, D. S. S. B., Kumari, T. D., & Vikas, B. (2018). Application of machine learning classification algorithms on hepatitis dataset. *International Journal of Applied Engineering Research*, 13(16), 12732–12737.

29. Nancy, P., Sudha, V., & Akiladevi, R. (2017). Analysis of feature selection and classification algorithms on hepatitis data. *International Journal of Advanced Research in Computer Science Engineering and Information Technology*, 6(1), 19–23.
30. Aravind, A., Bahirvani, A. G., Quiambao, R., & Gonzalo, T. (2020). Machine learning Technology for Evaluation of liver fibrosis, inflammation activity and steatosis (LIVERFASTM). *Journal of Intelligent Learning Systems and Applications*, 12(2), 31–49.
31. Murty, S. V., & Kumar, R. K. (2019). Enhanced classifier accuracy in liver disease diagnosis using a novel multi-layer feed forward deep neural network. *International Journal of Recent Technology and Engineering*, 8, No.2.
32. Bansal, H., Sharma, K., & Kaur, G. (2020). A hybrid approach for the prediction of liver disorder using data mining. *International Journal of Application or Innovation in Engineering and Management*, 9(7), 164–171.
33. Kadu, G., Raut, R., & Gawande, S. S. (2018). Diagnosis of liver abnormalities using support vector machine. *International Journal for Research Trends and Innovation*, 3(7), 2018.
34. Vishwanath, S., Ankita, K., Ajay, S., Maity, A., & Kumara, B. A. (2010). Prediction and diagnosis of liver disease. *International Journal of Advance Research, Ideas, and Innovations in Technology*, 6(3), 1–3.
35. Rajeswari, P., & Reena, G. S. (2010). Analysis of liver disorder using data mining algorithm. *Global Journal of Computer Science and Technology*, 10(14), 48–52.
36. Gogi, V. J., & Vijayalakshmi, M. N. (2018, July). Prognosis of liver disease: Using machine learning algorithms. In *In 2018 International Conference on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIEECE)* (pp. 875–879). IEEE.
37. Kuppan, P., & Manoharan, N. (2017). A tentative analysis of liver disorder using data mining algorithms J48, decision table and naive Bayes. *International Journal of Computing Algorithm*, 6(1), 2239–2278.
38. Juliet, P. L., & Tamildelvi, P. R. (2017). A comprehensive analysis on pre-processing and classification technique in data Mining for Predicting Liver Disorder from USA liver patient data. *International Journal of Advance Research in Science and Engineering*, 6, No.12.
39. Ramalingam, V. V., Pandian, A., & Ragavendran, R. (2018). Machine learning techniques on liver disease A survey. *International Journal of Engineering & Technology*, 7(4.19), 485–495.
40. Thangaraju, P., & Mehala, R. (2015). Novel classification-based approaches over cancer diseases. *System*, 4, 294–297.
41. Jangir, S. K., Joshi, N., Kumar, M., Choubey, D. K., Singh, S., & Verma, M. (2021). Functional link convolutional neural network for the classification of diabetes mellitus. *International Journal for Numerical Methods in Biomedical Engineering*, 37, e3496.
42. Choubey, D. K., Tripathi, S., Kumar, P., Shukla, V., & Dhandhanian, V. K. (2021). Classification of diabetes by kernel based SVM with PSO. *Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science)*, 14(4), 1242–1255.
43. Choubey, D. K., Kumar, M., Shukla, V., Tripathi, S., & Dhandhanian, V. K. (2020). Comparative analysis of classification methods with PCA and LDA for diabetes. *Current Diabetes Reviews*, 16(8), 833–850.
44. Choubey, D. K., Kumar, P., Tripathi, S., & Kumar, S. (2020). Performance evaluation of classification methods with PCA and PSO for diabetes. *Network Modeling Analysis in Health Informatics and Bioinformatics*, 9(1), 1–30.
45. Choubey, D. K., Paul, S., & Dhandhanian, V. K. (2017). Rule based diagnosis system for diabetes. *An International Journal of Medical Sciences*, 28(12), 5196–5208.
46. Choubey, D. K., & Paul, S. (2017). GA_RBF NN: A classification system for diabetes. *International Journal of Biomedical Engineering and Technology*, 23(1), 71–93.
47. Choubey, D. K., & Paul, S. (2016). Classification techniques for diagnosis of diabetes: A review. *International Journal of Biomedical Engineering and Technology*, 21(1), 15–39.
48. Choubey, D. K., & Paul, S. (2017). GA_SVM: A classification system for diagnosis of diabetes. In *Handbook of Research on Soft Computing and Nature-Inspired Algorithms* (pp. 359–397). IGI Global.

49. Bala, K., Choubey, D. K., Paul, S., & Lala, M. G. N. (2018). Classification techniques for thunderstorms and lightning prediction: A survey. In *Soft-Computing-Based Nonlinear Control Systems Design* (pp. 1–17). IGI Global.
50. Choubey, D. K., Paul, S., Bala, K., Kumar, M., & Singh, U. P. (2019). Implementation of a hybrid classification method for diabetes. In *Intelligent Innovations in Multimedia Data Engineering and Management* (pp. 201–240). IGI Global.
51. Rawal, K., Parthvi, A., Choubey, D. K., & Shukla, V. (2021). Prediction of leukemia by classification and clustering techniques. In *Machine Learning, Big Data, and IoT for Medical Informatics* (pp. 275–295). Academic.
52. Choubey, D. K., Paul, S., Kumar, S., & Kumar, S. (2017, February). Classification of Pima indian diabetes dataset using naive bayes with genetic algorithm as an attribute selection. In *Communication and computing systems: Proceedings of the international conference on communication and computing system* (pp. 451–455). ICCCS.
53. Bala, K., Choubey, D. K., & Paul, S. (2017, April). Soft computing and data mining techniques for thunderstorms and lightning prediction: A survey. In *In 2017 International Conference of Electronics, Communication and Aerospace Technology (ICECA)* (Vol. 1, pp. 42–46). IEEE.
54. Choubey, D. K., Paul, S., & Dhandhanian, V. K. (2019). GA_NN: An intelligent classification system for diabetes. In *Soft Computing for Problem Solving* (pp. 11–23). Springer.
55. Kumar, S., Mohapatra, U. M., Singh, D., & Choubey, D. K. (2020). IoT-based cardiac arrest prediction through heart variability analysis. *Advanced computing and intelligent engineering. Proceedings of ICACIE 2018*, 2(2), 353.
56. Kumar, S., Mohapatra, U. M., Singh, D., & Choubey, D. K. (2019, May). EAC: Efficient associative classifier for classification. In *2019 International Conference on Applied Machine Learning (ICAML)* (pp. 15–20). IEEE.
57. Pahari, S., & Choubey, D. K. (2020, February). Analysis of liver disorder using classification techniques: A survey. In *In 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)* (pp. 1–4). IEEE.
58. Parthvi, A., Rawal, K., & Choubey, D. K. (2020, July). A comparative study using machine learning and data mining approach for Leukemia. In *In 2020 International Conference on Communication and Signal Processing (ICCS)* (pp. 0672–0677). IEEE.
59. Sharma, D., Jain, P., & Choubey, D. K. (2020, July). A comparative study of computational intelligence for identification of breast cancer. In *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences* (pp. 209–216). Springer.
60. Srivastava, K., & Choubey, D. K. (2019, December). Soft computing, data mining, and machine learning approaches in detection of heart disease: A review. In *International Conference on Hybrid Intelligent Systems* (pp. 165–175). Springer.
61. Choubey, D. K., Mishra, A., Pradhan, S. K., & Anand, N. (2021, June). Soft computing techniques for dengue prediction. In *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 648–653). IEEE.
62. Bhatia, U., Kumar, J., & Choubey, D. K. (2022). Drowsiness image detection using computer vision. In *Soft Computing: Theories and Applications* (pp. 667–683). Springer.
63. Pahari, S., & Choubey, D. K. (2022). Analysis of liver disorder by machine learning techniques. In *Soft Computing: Theories and Applications* (pp. 587–601). Springer.
64. Choubey, D. K., Paul, S., & Bhattacharjee, J. (2014). Soft computing approaches for diabetes disease diagnosis: A survey. *International Journal of Applied Engineering Research*, 9(21), 11715–11726.
65. Choubey, D. K., & Paul, S. (2015). GA_J48graft DT: A hybrid intelligent system for diabetes disease diagnosis. *International Journal of Applied Engineering Research*, 7(5), 135–150.
66. Choubey, D. K., & Paul, S. (2016). GA_MLP NN: A hybrid intelligent system for diabetes disease diagnosis. *International Journal of Bio-Science and Bio-Technology*, 8(1), 49.

Artificial Intelligence Based Transfer Learning Approach in Identifying and Detecting Covid-19 Virus from CT-Scan Images



Soubraylu Sivakumar, D. Haritha, Ratnavel Rajalakshmi, S. Shanmugan, and J. Nagaraj

Abstract A virus that influences the world since late 2019 is Coronavirus (Covid-19). The WHO has given preliminary guidelines to be followed by every individual in this pandemic situation. Even then many countries are facing a lot of consequences and human loss. Many countries tried to produce different vaccines in the market for this covid-19. But, none of these vaccines is 100% immune against this virus. Early detection of this virus from a victim will provide a faster recovery and good health. One such solution can be provided through the determination of covid-19 from Computerized Tomography (CT) scanned images. A strong and effective machine learning approach is needed in this pandemic situation to locate, stop, and control the spread of COVID-19. To extract the complex features and to provide a better classifier from these high-dimensional CT scan images an effective feature extractor and optimal classifier are required. A transfer learning

S. Sivakumar (✉)

Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

e-mail: sivas.postbox@gmail.com

D. Haritha

Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

e-mail: haritha_donavalli@kluniversity.in

R. Rajalakshmi

School of Computing, Vellore Institute of Technology, Chennai, India

e-mail: rajalakshmi.r@vit.ac.in

S. Shanmugan

Research Centre for Solar Energy, Department of Engineering Physics, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

e-mail: s.shanmugam1982@gmail.com

J. Nagaraj

Computer Science and Engineering, Madanapalle Institute of Technology and Science, Madanapalle, Andhra Pradesh, India

e-mail: nagrajan31@gmail.com

based machine learning method is proposed named as VGG16-SVM. At the top of this proposed model, a multiple stacked smaller size kernel named VGG16 is used to select the complex features that are hidden from the CT-Scan images. These reduced features are fed into a kernel-based Support Vector Machine (SVM) for analysis. Three distinct experiments are explored to determine the best model in the identification of Covid-19 which includes VGG16-SVM, VGG16-Random Forest and VGG16-XGBOOST. The experiments are conducted on 348 CT-Scan images. Image data augmentation strategies are employed to enhance the proposed model's accomplishment and prevent overfitting. The metrics used in our research work are training and validation accuracy. From the experimental analysis, we found that applying VGG16-SVM on CT scanned images obtained a best performance result in recognizing the Coronavirus.

Keywords VGG16 · Support Vector Machine · Deep convolutional neural network · Covid-19 · CT-Scan images · Transfer learning · Random Forest · XGBoost

1 Introduction

A novel virus named Covid-19 has out broken from china in December 2019 from Wuhan city. It has infected globally a large unit of populations regardless of their community, race and location. People affected with a disease like cancer, diabetes, chronic respiratory and cardiovascular are affected severely by this SARS-CoV-2. It badly strikes the respiratory canal, makes a lesion and affects the regular functionality of the lungs. The symptoms of this epidemic can be identified as tiredness, dry cough, respiratory illness, fever and loss of sensation of taste.

There are 591,683,619 COVID-19 proved cases around world. This epidemic caused 6,443,306 human losses in the world on 23 August 2022 [30]. To reduce human loss, more tests are to be performed per million populations to identify this disease and to know the spreading rate of the epidemic. Covid-19 infects and creates an impact on the respiratory organ of humans. So, it will be easy to detect Covid-19 rapidly through medical imaging of the chest. A Computer Tomography Scan of a Coronavirus victim is the simplest and easy method to quickly detect and automatically [16] diagnose Covid-19.

The motivation of this study is to automatically detect Covid-19 from patient CT scan images [11]. A collection of 348 CT-Scan images from a senior radiologist is used in this experiment. These scanned images are taken from 216 COVID-19 patients. A data augmentation method that magnifies a particular image from the dataset is added to optimise the model's performance. In the augmentation process, the 300 training and 100 validation images are expanded into 600 training and 200 validation images.

Two methods are analyzed in the various experiments to find an effective model for detecting Covid-19 namely, (i) Cross Validation and (ii) the Learning Curve

method. Both these methods are applied to three sets of experiments viz., (i) VGG16 with Support Vector Machine (VGG16-SVM), (ii) VGG16 with Random Forest (VGG16-RF) and (iii) VGG16 with eXtreme Gradient Boost (VGG16-XGBoost). In the folding method, each fold is represents an experiment. So, a total of 10 experiments are conducted in 10 fold method. In this method, VGG16-SVM have achieved a mean Accuracy of 87.75% compared with the other two models. In the learning curve method, the no. of training samples is increased to the multiple of 27. A total of 10 experiments are conducted with a training size of 270. The VGG16-SVM has obtained an Accuracy of 84.66% while compared with VGG16-RF and VGG16-XGBoost.

Each block of VGG16 is composed of 2 convolution layers with a receptive size of 3×3 along with a ReLu activation and a max pooling layer of 2×2 -pixel window. This neural network is composed of five such blocks in a series. This complex non-linear layer helps to extract a latent rich feature from the high dimensional space. These features are fed into the SVM. The quadratic programming of SVM used to perform iterative calculation on the input until a feasible solution is obtained. It scales well for sparse data like images, especially for binary classification problems. It has attained good results for both 10-fold cross-validation and linear curve method while compared with VGG16-RF or VGG16-XGBoost models. The contribution in this article comes in three folds.

1. The complex chronological features from the high dimension CT-Scan images are extracted with a sequence of multiple non-linear layers from VGG-16,
2. The rich extracted features from CT-Scan are gave into the SVM an iterative computation sequential optimizer classifier for classification and
3. Using a Transfer Learning (TL) approach, the deep CNN and a conventional classifier are integrated to provide a VGG16-SVM model to classify and detect Covid-19 patients.

The entire research work structure is as follows. The existing work limitations are discussed in Sect. 2. In Sect. 3, the proposed methodology, the feature selection method, the data augmentation method and the different image classifiers are discussed. In Sect. 4, the dataset, metrics, experimental setup and experimental results are explained. The experiment results are concluded with future enhancement in Sect. 5.

2 Related Work

Covid-19 [1] is asserted as a pandemic disease by WHO. The Covid-19 disease affects the lungs of humans and lung abnormality is the well-known diseases among all ages of human being. Due to lung infection, the respiratory system is also damaged. Image modeling detects the infection severity. The scanned images of the patient are helpful in detecting the covid-19 through a three phase model and the steps involved is as follows,

Phase 1: Data accession uses stagnant wavelets

Phase 2: CNN model is used for detection of covid-19

Phase 3: This phase uses CT scan images. It uses well known pre-trained architecture [18] of ResNet Series such as 18, 50 and 101 including SqueezeNet for assessment.

In this classification process, 70 and 30% of images are used in training and validation. The performance is calculated by computing the common performance measures. The ResNet18 uses TL approach is a pre-trained model used in evaluating the experiments.

For image analysis, the deep learning methodology [2] is greatly expanded. Similar to how deep learning applications (DL) to medical images of the respiratory system have emerged, these trials are displaying great potential. The development of DL applications for medical image analysis with a focus on lung image analysis is discussed in order to better understand the achievements to COVID-19. Deep learning tasks like segmentation, categorization, and recognition [19] as well as abnormal lung pathologies like respiratory illnesses, lung cancer, COVID-19, as well as other infections are emphasised in this survey that has 160 contributions. It also provides an overview of the current pandemic situation.

Before universal testing (CT imaging and PCR) prior to surgery of 72–120 h blood tests and anesthesiology related to imaging and surgical procedure respectively are scheduled [3, 4]. Of the 211 emblematic victims who have been tested ahead surgery, six trusted positive PCR findings. One Hundred Four patients who underwent elective surgery as part of the current study were healthy both before and after the procedure. Out of 336 patients who were cancelled, only 12% agreed to a new appointment time right away. This led to a 70% decrease in elective surgery and a 50% decrease in arthroplasties following lockdown. The post-COVID period did not see an increase in the complication rate. Patients struggle to comprehend the new guidelines founded by health organizations and have unclear ideas about screening.

In two orthopaedic surgery centres, 1397 patients underwent chest CT scans between March 1 and May 10, 2020, during the peak of the epidemic [3, 4]. We chose 118 patients out of 1397 for orthopaedic surgical treatment who displayed trauma symptoms. Thirty nine patients out of 118 were subjected to PCR testing to investigate COVID-19 infection. The Chest CT scan provides useful information for orthopaedic surgeons based on clinical status (symptomatic or non-symptomatic) and is graded from 0 (no value) to 3 (high value) (high value). Specific pathways and exposure to radiation were analyzed and discussed with existing possibilities. The evaluation of the increase in CT scanning during the COVID-19 pandemic led to the development of this result, which was based on prior treatment performed during the same time period (1st March 2018–15th April 2018). Total 118 patients are used for evaluation with 102 patients with negative and 16 patients with positive chest CT scan. Regarding the PCR results, the chest value is reported as having a sensitivity of 81%, a specificity of 93%, a positive predictive value of 86%, and a positive predictive value of chest CT ($p = 0.001$). The following are the grades (1 for 71

cases, 2 for 5 cases, 3 for 11 cases). Only 2% of the CT scans performed during the pandemic period were for orthopaedic or trauma patients, which accounted for 20% of the total. This outcome was ten times greater when compared to the preceding control period.

It is difficult to identify Covid-19 while looking at clinical photographs of patients [5]. The TL method has been employed in the clinical imaging of a variety of lung conditions, which include covid-19 [10]. Similar condition that predated covid-19 is pneumonia lung disease. We must put forth a new model to stop the disease from spreading in order to anticipate COVID-19. We can learn that COVID-19 and viral pneumonia are the same thanks to the TL approach. Due to noise burden outside of tissues and lesions, it is difficult to see aberrant features in images. Texture characteristics are extracted employing haralick functionalities, which target solely on the area of attraction.

The deep learning method was trained using 120 chest CT scans with pulmonary infiltrates [6]. The regions and arteries of the lungs are segmented using this technique. Using 72 consecutive scans from 24 COVID-19 individuals, this method determines the existence and evolution of infiltrates connected to COVID-19. The components of this method are as follows: (1) Computerized lung borders and artery edge detection; (2) Lung frontiers validation among scans; (3) Digital recognition of the Digitized Pneumonitis Zones; and (4) Evaluation of Cancer Progression. The understanding among the regions that the radiologist manually outlined and the regions that the computer detected was evaluated using the Dice coefficient. A heat map illustrating the difference between scans was produced using serial scans that were enrolled and recorded. Using a five-point Likert scale, two radiologists evaluated the accuracy of the heat map used to indicate progression. They had a Dice coefficient of 81% (CI 76–86%), which indicated understanding among computer identification and human delineation of pneumonic regions. The algorithm had an 84% (CI 81–86%) specificity and a 95% (CI 94–97%) sensitivity. In order to detect significant pneumonia zones, radiologists assessed 95% (CI 72–99) of heat maps as at least “adequate” for portraying disease progression.

The world is anticipating what the novel coronavirus [7] can do to humanity because it has a number of distinctive properties. It is essential to identify and isolate the ill patients as soon as practical because a coronavirus-specific vaccination is not yet available. There aren't as many testing facilities and tools available as we had anticipated. This study paper discusses the use of machine learning techniques for gaining crucial insights. A lung CT scan should be the first obtainable test for real-time reverse transcriptase-polymerase chain reaction (RT-PCR). Patients with COVID-19 pneumonia seem to be difficult to differentiate from those who have other viral pneumonias on a global scale [9]. For training and testing, Microsoft Azure's custom vision software is built on machine learning methods. With the suggested method, COVID-19 classification accuracy is 91%.

Our goal [8] is to compare the diagnostic efficacy of a 30-mAs chest CT protocol with a 150-mAs standard-dose regular procedure using imaging of COVID-19 pneumonia. After the IRB approved the trial, COVID-19 participants 50 years of age or older were sent for chest CT scans and received normal CXRs the same

day. A standard dosage (150 mAs) chest CT scan was performed first. Low-dose CT (30 mAs) for COVID-19 identified individuals was carried out right away. The accuracy of CTs with low and high doses was distinguished. The 20 patients in this study had an average age of 64.20 ± 13.8 . The results were: low-dose 1.80 ± 0.42 mSv and standard-dose 6.60 ± 1.47 mSv. Absolute cancer risk per mean cumulative effective dose values as follows standard-dose 2.71×10^{-4} and low-dose 0.74×10^{-4} .

2.1 Existing Limitations

A major virus that influences the world from the year 2020 is named Coronavirus (COVID-19). Identifying the Covid-19 patient immediately is a great challenge in these pandemic situations. The various approaches and restrictions for identifying Covid-19 employing CT-Scan images are presented in Table 1. Some of the inferences we come across in this literature review limitations are:

- (i) Not able to learn deep features,
- (ii) It causes overfitting and overemphasizes outliers.
- (iii) It needs feature scaling and
- (iv) It is time-consuming and requires more computational power.

To address all the above issues, a robust model is required to provide a better result. The proposed VGG16-SVM model addresses some of the limitations listed above and its performance is discussed in the below section.

Table 1 List the existing methodology and limitations of identification of Covid-19 from CT-Scan images

S.No.	Author	Methodology	Limitations
1.	M. Loey et al. [32]	ALexnet	It struggles to scan for all features and its performance is poor in most of the use cases
2.	V. Perumal et al. [5]	Inception V3	It is time-consuming and requires more computational power
3.	G. Muscas et al. [25]	Gradient boosting machine	It causes overfitting and overemphasize for outliers
4.	Xiangjun Wu et al. [26]	Multi-View deep fusion model	It is time-consuming and requires more computational power
5.	H. Yasar and Ceylan [29]	Local Binary Pattern-KNN 10 fold cross validation	Does not scale well with high dimensions and needs feature scaling
6.	V. Shah et al. [33]	CTnet-10	Not able to learn deep features

3 Materials and Methods

The methodology used, image pre-processing method, data augmentation method, feature weighting method, proposed architecture, machine learning methods and multivariate analysis are discussed in this section.

3.1 Methodology

The CT scan images from the datasets are applied to this deep neural network design by Ganesan et al. [12]. There are 348 Covid and 348 Non-Covid CT Scan pictures in the original dataset. In order to increase the quantity of photos and prevent the results from being overfit, a data augmentation approach is used in this process. The quantity of photographs in each category has increased to 52 after the augmentation process (i.e., 400 images). Three categories of augmented images—training, validation, and testing—are used. The training, validation, and testing sets each comprise 200, 100, and 100 photos after the images have been divided. With the help of the VGG-16 deep neural network, the characteristics are extracted. It is a 16 layer neural network that assists in extracting significant essential features from the Covid and Non-Covid CT-Scan dataset [15].

The SVM, RF, and XG Boost are three machine learning models whose hyperparameters are selected using the grid search approach. The classifiers’ ideal parameters are set with the use of grid search. In order to forecast the testing set, many classifiers are learned. To forecast the model’s performance outcome, two techniques are used. When predicting CT-Scan images using various classifiers, K-Fold cross-validation and learning curve approach are utilised. The effectiveness of the various classifiers is evaluated using accuracy as a criterion. Figure 1 displays the process used to predict Covid CT-Scan results.

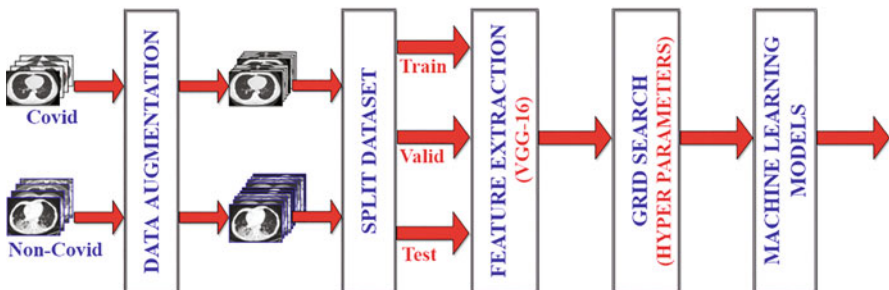


Fig. 1 Methodology used in the prediction of Covid CT-Scan images

3.2 Preparing Dataset

The *path* variable holds the folder path of all the files to be renamed. The function *listdir()* returns the list of all files, while *enumerate()* function helps to iterate over each file in the folder. In every loop iteration, the *count* takes a value from 0 to $n - 1$, where n is the number of files in the folder. The *dst* variable holds the file name as *covid0.png* and it helps to rename the file. The source and destination path names of the file are stored in the variables *src* and *dst*. The function *rename()* helps to rename the file from source name to destination name. Base directory and train folder are combined with the *join()* method and stored in *train_dir* variable. The function *mkdir()* creates a new folder for training images to be placed. In similar way, a Covid and Non-Covid folder will be created inside the *train* folder. The same operations are continued for creating *validation* and a *test* folder for storing the images. *Shutil* is a library that copies a file from the original folder to the newly created folder. The *src* represents the source of the file to be copied and the *dst* represents the destination of the file to be placed.

3.3 Data Augmentation

A sample instance of Covid and Non-Covid CT-Scan image is shown in Fig. 2. There are 348 images in each category of this dataset. To increase the performance and avoid overfitting of the model, we have included an image data augmentation step before feature extraction [23]. Figure 2a, b shows the Covid and Non-covid MRI Image respectively. *ImageDataGenerator* is a class responsible to define an object that is required for augmentation. There are several methods available for augmentation and a few of them are flipping, rotating, shearing, cropping and zooming. A zoom operation is applied to the images as an augmentation step in

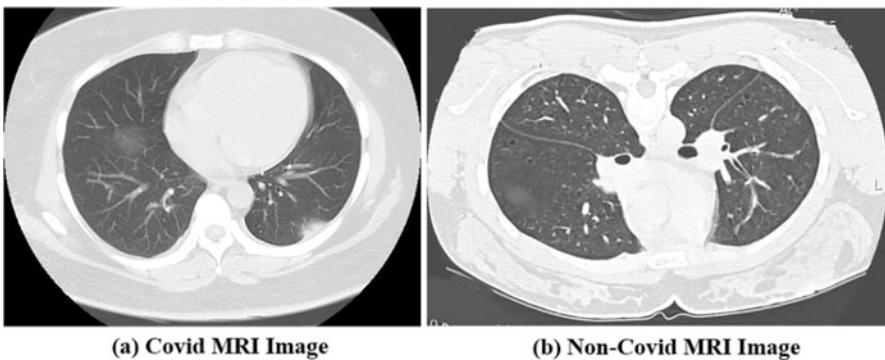


Fig. 2 Displays a sample instance from Covid and Non-Covid CT-Scan images before augmentation

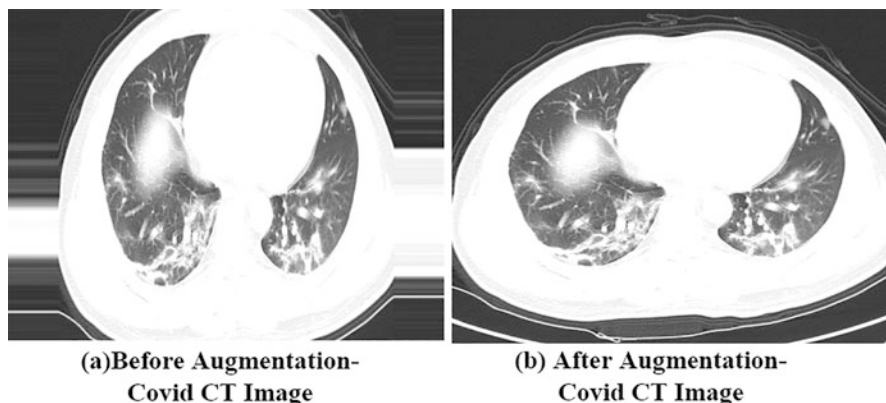


Fig. 3 Sample instance from Covid CT-Scan images before augmentation and after augmentation

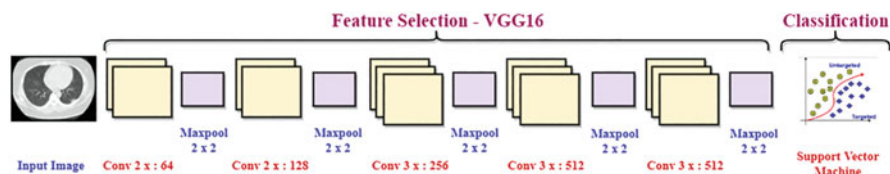


Fig. 4 Proposed architecture VGG16 with Support Vector Machine (VGG16-SVM)

our preprocessing. Images are loaded from the folder using `load_img()` function and the image are converted into 3D Numpy array using `img_to_array()` function. The `reshape()` method changes the dimension of the Numpy array.

The method `flow()` is responsible for applying the zoom effect to the image, saving the image in a specific folder and specific format with an added prefix file name as “*aug*”. Figure 3a, b shows an sample instance from Covid CT-Scan images before augmentation and after augmentation respectively. After augmentation, the CT scan image dataset contains 400 images in each category of Covid and Non-Covid set. Fifty two images are added by the augmentation process to each category of the dataset.

3.4 Proposed Architecture

VGG16 is used at the top of the proposed architecture, and SVM is used at the bottom. It is depicted in Fig. 4. The VGG16 extracts the complex feature from the high dimensional space and it is fed into the SVM for classification. The SVM is a more suitable classifier for the high dimensional vector without any preprocessing and provides good results.

3.4.1 VGG16

VGG16 is employed to extract the features from the images [21]. The deep neural network is made up of the first five blocks in the architectural diagram. VGG16 is composed of 2 two layer and 3 three layer convolution layer. In between the convolution layer a maxpooling layer is used to choose the prominent features. These five blocks constitutes a 16-layer VGG, with an input shape (width, height, channel) as (224, 224, 3). In VGG-16, the top three fully connected are made as false, to convert this deep neural network to act as a feature extractor instead of an image classifier. The output of the fifth block is provided as an input to well-known machine learning techniques like SVM, RF, and XG Boost rather than a fully linked layer. A method from the *Keras* library named *summary()* is used to display the entire structure of the VGG16 neural network and the definition of VGG16.

3.4.2 Support Vector Machine

Given a set of data $D = \{x_i, y_i\}_{i=1}^n$ with $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$, let us take for instant the data points are linearly separable. Formerly we can define, that all data points marked $y_i = -1$ lie on one side ($h(x) < 0$) and other data points marked as $y_i = +1$ lie on another side ($h(x) > 0$). The basic idea of SVMs is to take the canonical hyperplane, defined by the weight vector ‘w’ and the bias ‘b’, that gives the maximum margin among all potential classifying hyperplanes $h(x) \equiv w^T x + b = 0$. If δ_h^* presents the margin for hyperplane $h(x) = 0$, then the goal is to determine the optimum hyperplane h^* and it is given in the Eq. (1).

$$h^* = \arg \max_h \{ \delta_h^* \} = \arg \max_{w,b} \left\{ \frac{1}{\|w\|} \right\} \quad (1)$$

The SVM task is to determine the hyperplane that enlarges the margin $\frac{1}{\|w\|}$, subject to the ‘n’ constraints, namely, $y_i (w^T x_i + b) \geq 1$ for all data points $x_i \in D$.

Rather than enlarging the margin $\frac{1}{\|w\|}$, we obtain an equivalent expression if we minimize $\|w\|$. In fact, we can obtain an equivalent minimization expression given as follows in Eq. (2)

$$\begin{aligned} \text{Objective Function : } & \min_{w,b} \left\{ \frac{\|w\|^2}{2} \right\} \\ \text{Linear Constraints : } & y_i (w^T x_i + b) \geq 1, \forall x_i \in D \end{aligned} \quad (2)$$

It is a supervised machine algorithm [13] used for both regression and classification purposes. The wide functionality of SVM suits this method for classification problems. The features of each data point in this algorithm are plotted in n-dimensional space. Each coordinate point represents a feature in the problem space. Then, classification is done by distinguishing between the two classes with a hyperplane. This classifier is best suitable for discriminating the two classes. It

works well in separating the two classes with a margin. This method is suitable for images, videos and audios which are in high dimensional nature. In the decision function of SVM, a subset of training points is used that effectively uses the memory during the classification.

3.5 *Machine Learning Methods*

The machine learning methods used in the classification of the CT-Scan images are discussed below:

3.5.1 XGBoost

It is otherwise called Extreme Gradient Boost. It is a decision tree based model [20] designed to increase in speed of training and improve the model performance. The algorithm is designed to optimize using the memory for situations like distributed computing, cache optimization, out-of-core optimization and parallelization. In general, the gradient boosting model corrects the error of the new model from features learned from the prior model. This method supports the prediction problems through classification and regression. It can handle automatically the missing values in any instance of the dataset. This constructs the tree parallel and increases the computation speed. In the training phase, the model is designed to fit any new data.

3.5.2 Random Forest

The important feature of the Random Forest is the ease of use, robustness and accuracy. Two important feature selection methods are included in this algorithm mean decrease accuracy and mean decrease impurity. In the mean decrease accuracy method, feature values are permuted to decrease the model accuracy. Important variables have significant influences in decreasing the accuracy of the model [14], while unimportant variables do no effect on accuracy. In another method, feature that decreases weighted impurity of the tree are considered for the training. For classifications, Information Gain or Gini impurity and for regression, variances are used in feature computations. Decreases in each feature impurity are averaged and this measure is used to rank those features.

3.6 *Multivariate Analysis*

The multivariate analysis deals with images that have more than one measurement per pixel. High dimensional data will be analyzed using principle component

analysis (PCA) to simplify the process while conserving patterns and trends. We have adopted the image reconstruction technique using PCA to capture more information and variance from the data. For that purpose, we have taken a Covid CT scan image from the dataset and the image reconstruction method using PCA is applied. Figure 5a, b shows an original instance from Covid CT-Scan image and reconstructed Covid CT-Scan image using PCA with $k = 25$ respectively. A line chart is shown in the Fig. 6 with a number of components and cumulative explained variance in 'x' and 'y' respectively. From the chart, it is clear that 25 'k' components are required with a cumulative variance of 95% for successful image reconstruction.

From the chart, we have observed that the number of principal components makes a difference in the reconstruction of the image. We also plotted different subplots to compare the relative difference between the number of 'k' components and as shown in the Fig. 7. Although the image clarity has modified, the suggested model's

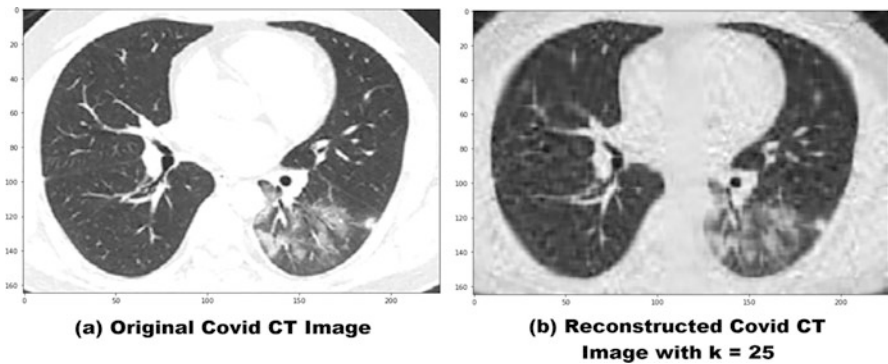


Fig. 5 An original instance from Covid CT-Scan image and reconstructed Covid CT-Scan image

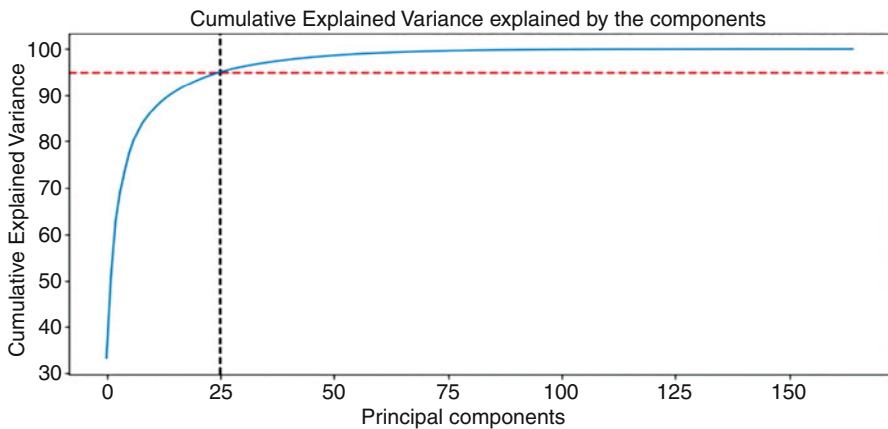


Fig. 6 A line chart with cumulative variance and 'k' principal components

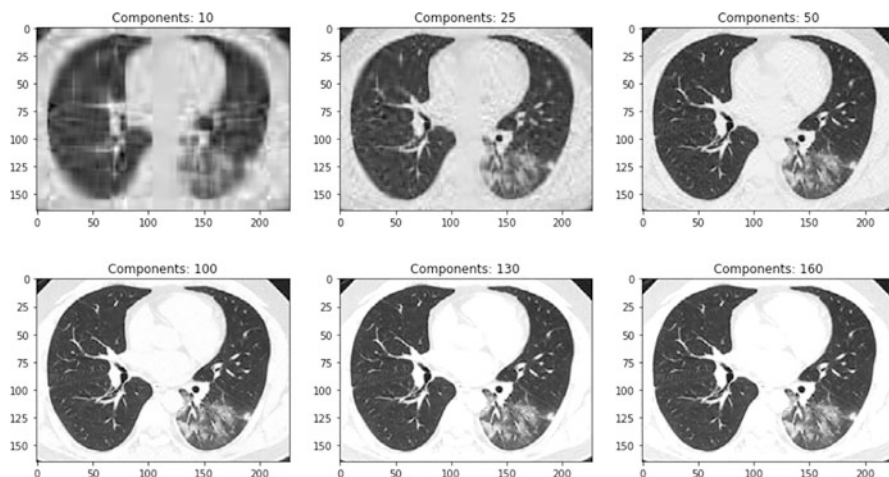


Fig. 7 Subplots of Covid CT-Scan Image for different ‘k’ principle components using reconstruction approach

overall accuracy has not improved as a result of the multivariate approach. However, the increase in image reconstruction pixel size results in an increase in computational time.

4 Results and Discussion

This section deals with the software and hardware specification, the dataset used for the analysis of Covid-19, metrics used for evaluation, hyperparameter tuning of the model, experimental results and comparative analysis of existing with proposed models are discussed.

4.1 Experimental Setup

On the CT-Scan X-ray images from the Covid dataset, a comparison analysis of the three TL methods is done [24]. For the experimental configuration, we used a computer with an Intel Core i3 8th generation, 12 GB of RAM, and a 1 TB hard drive running Windows 10. A top radiologist at Tongji Hospital in Wuhan, China provided this dataset. There are 349 Covid-19 CT scan images in this collection. Tensorflow 1.14, Keras 2.2.4, and Python 3.7 are all used to carry out the tests. The features are extracted from the images by employing the VGG-16 DL model. The VGG-16 is a predefined model loaded from the Keras application. Three experiments are carried on the Covid dataset. It includes viz., SVM, RF and XG Boost [31]. A TL approach is applied to the extracted feature with the above given traditional machine learning methods.

4.2 Dataset and Metrics

The dataset is collected from Tongji Hospital, Wuhan, China between the period January 2020 and April 2020. It contains a collection of 348 CT images aggregated from 216 patients. The dataset is taken from Github. CT scan images are otherwise called Computerized Tomography or CAT scan images. These images are aggregated from Covid victims by the senior radiologist. These images are helpful in the diagnosis of Covid patients. Quick treatment can be given with the help of a better and more efficient image diagnosis procedure through deep learning methods.

Each Covid and normal victims have 348 CT scan images. It is divided into 250, 50 and 48 images for training, validation and testing sets respectively and it is shown in Fig. 8. Image augmentation is a method or technique [17] that expands the capacity of the dataset to prevent overfitting issues. It also helps to increase or ameliorates the functioning of the model. After image augmentation, the dataset consists of 300 and 100 images for training and validation set in each category of Covid and Non-Covid dataset and it is shown in Fig. 9. The augmentation dataset is prepared by zooming on the original dataset with the help of *ImageDataGenerator()*. The accuracy measures the goodness of a model in predicting the result and it is given in the Eq. (3). In this equation, the TP, FP, TN and FN represent True Positive, False Positive, True Negative and False Negative respectively.

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

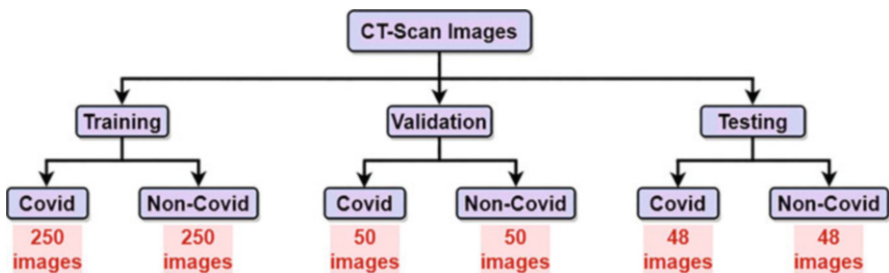
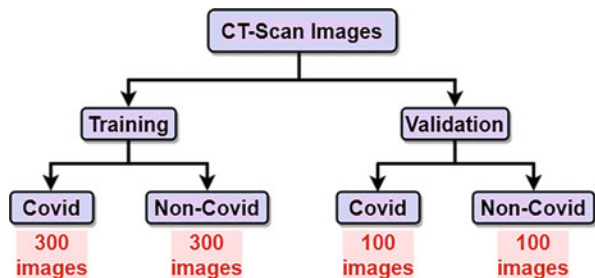


Fig. 8 Categorization of covid and non-covid CT-Scan images before augmentation

Fig. 9 Categorization of covid and non-covid CT-Scan images after augmentation



4.3 *Hyper Parameter Selection*

Grid search is a technique that helps to choose/compute parameters of a model that enables us to produce accurate predictions in the classification. It is also called an estimator that helps to perform optimal hyperparameter selection for a given model. This search helps us to maximize the prediction. The hyperparameters of the Random Forest classifier for the parameters random state, max. features, no. of estimators, max. depth and criterion are 42, 'log2', 200, 8, and 'entropy' respectively. The parameters that are set for the XGBoost classifier are small fractions of columns (0.7), learning rate (0.05), max. depth (6), min. sum of instance weight (11), missing input value (-999), no. of estimator (5) and no. of thread (4). To obtain optimal results in the prediction of Covid-19 victims, the SVM classifier hyperparameters are tuned as penalty (12), c (0.01), loss (squared_hinge) and kernel (linear).

4.4 *Experimental Results*

To find the effectualness of the proposed architecture in finding the Covid-19 patients from CT-Scan figures two methods are employed namely (i) Cross Validation method and (ii) Learning Curve method.

4.4.1 **Experiment 1: Cross Validation**

Conceptual Representation

The various types of cross-validation approaches are used in machine learning to validate the dataset. A few important types of cross-validation methods are (i) K fold Cross Validation, (ii) Stratified K fold Cross Validation, (iii) Leave P out cross validation and (iv) Time Series Cross Validation. A 10-fold cross-validation method is chosen to validate the CT scan images. The total size of the dataset is 400 images. The test set size used in each experiment of this cross-validation method is $400/10 = 40$ images. A Total of 10 experiments are conducted in this cross-validation. Each experiment [22] is representing one fold of the cross-validation. In experiment one, 1–40 images are assigned as the test set, while the remaining 41–400 images i.e., 360 images for the training set. Likewise, in experiment two, the training set is assigned with a range of 1–40 and 81–400 with a total of 360 images. The test set in experiments 1, 2, 3, ... 10 ranges from 1–40, 41–80, 81–120, ..., 361–400 respectively and it is shown in Fig. 10 with a yellow color. The sky blue color represents the training set used in the 10-fold cross-validation.

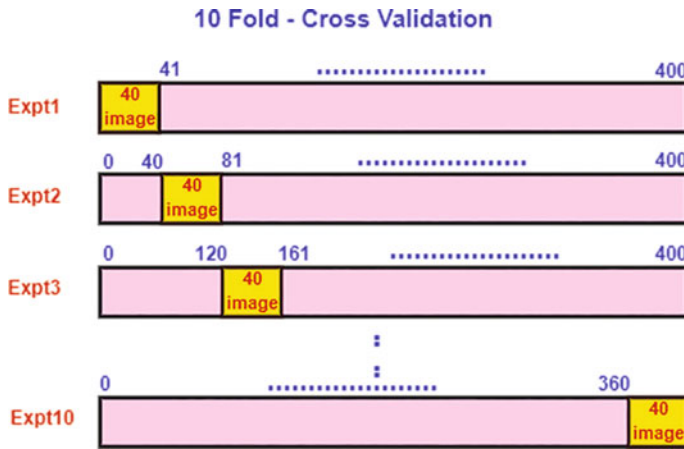


Fig. 10 Illustrates the conceptual representation of 10-Fold cross validation in CT scan images

Table 2 Experimental results of three TL models in terms of accuracy

	Expt1	Expt2	Expt3	Expt4	Expt5	Expt6	Expt7	Expt8	Expt9	Expt10
VGG16-SVM	92.5	90.0	90.0	82.5	90.0	85.0	92.5	87.5	82.5	85.0
VGG16-RF	75.0	77.5	87.5	77.5	87.5	87.5	82.5	82.5	80.0	82.5
VGG16-XGB	67.5	55.0	60.0	75.0	85.0	77.5	52.5	60.0	75.0	80.0

Results Discussion

Table 2 lists the accuracy results of three TL models in the order VGG16-SVM, VGG16-RF and VGG16-XGB. Ten experiments are conducted for 10-fold cross-validation in each model. The SVM obtained a maximum accuracy of 92.5% in experiment one, while XGBoost obtained a minimum value of 52.5% in the 7th fold cross-validation. The Mean accuracy of SVM, Random Forest and XGBoost models for 10-fold cross-validation is 87.75%, 82.00% and 68.75% respectively. SVM, Random Forest and XGBoost achieved a standard deviation of 3.61%, 4.30% and 10.73% respectively. A comparison of the three models is displayed in Fig. 11. Blue, Red and Green lines represent the SVM, Random Forest and XGBoost accuracy for cross-validation. SVM lies top in the experiment results of CT Scan images of the Covid19 dataset. The SVM classifier is composed of a function that is maximized in compliance with linear constraints on its variable using minimal consecutive optimization. This approach allows the identification of the viable solution by iterative computation, which generated an accuracy of 87.75%.

4.4.2 Experiment 2: (Learning Curve Method)

This method helps to monitor the various aspects of machine learning model performance and identify the lacking areas of the model. Thereby it helps to see the

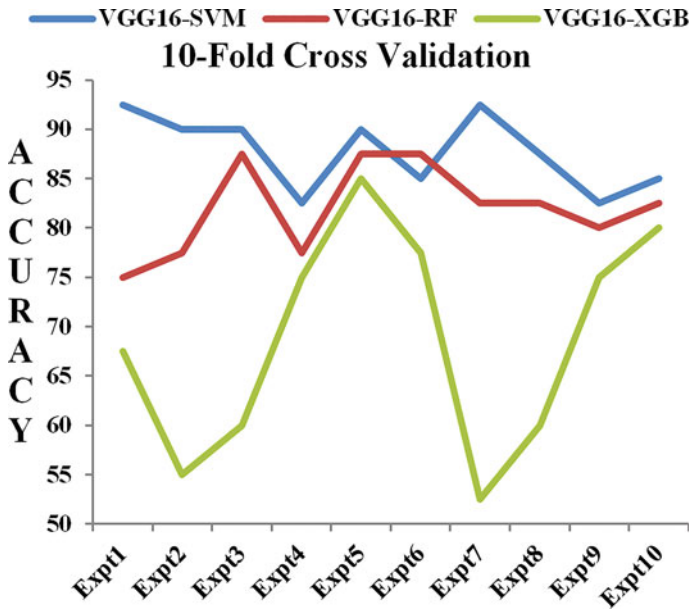


Fig. 11 Comparison of the three TL models in terms of accuracy

progress of the model, helps to build a plan to improve the performance of the model and helps to forecast the decision making process to be easy. The experimental results of the three TL models are compared between accuracy and the number of training samples as shown in Fig. 12. In the figure, (a), (b) and (c) represents the performance result of SVM, Random Forest and XGBoost respectively.

The validation accuracy of the model gradually increases as the training set size of the Covid dataset increases. Support Vector Machine obtained 70.66% accuracy at 27 training samples and 84.66% accuracy at 270 training images. SVM achieves the highest accuracy of 84.66% which is 6.33% and 15.33% greater than the Random Forest and XGBoost classifier respectively. The performance result of the SVM classifier is higher than the Random Forest and XG Boost for training samples. Table 3 shows the output of the learning curve method on the different classifiers. The Fig. 12a–c displays the performance of the learning curve method of SVM, Random Forest and XG Boost classifiers respectively. Random Forest and XGBoost are tree based classifiers, they need a special preprocessing step to handle. In this tree based method, output at each stage depends on the correlation between two trees in the forest and the strength of an individual tree. This enables us to overfit the two models for high-dimensional data.

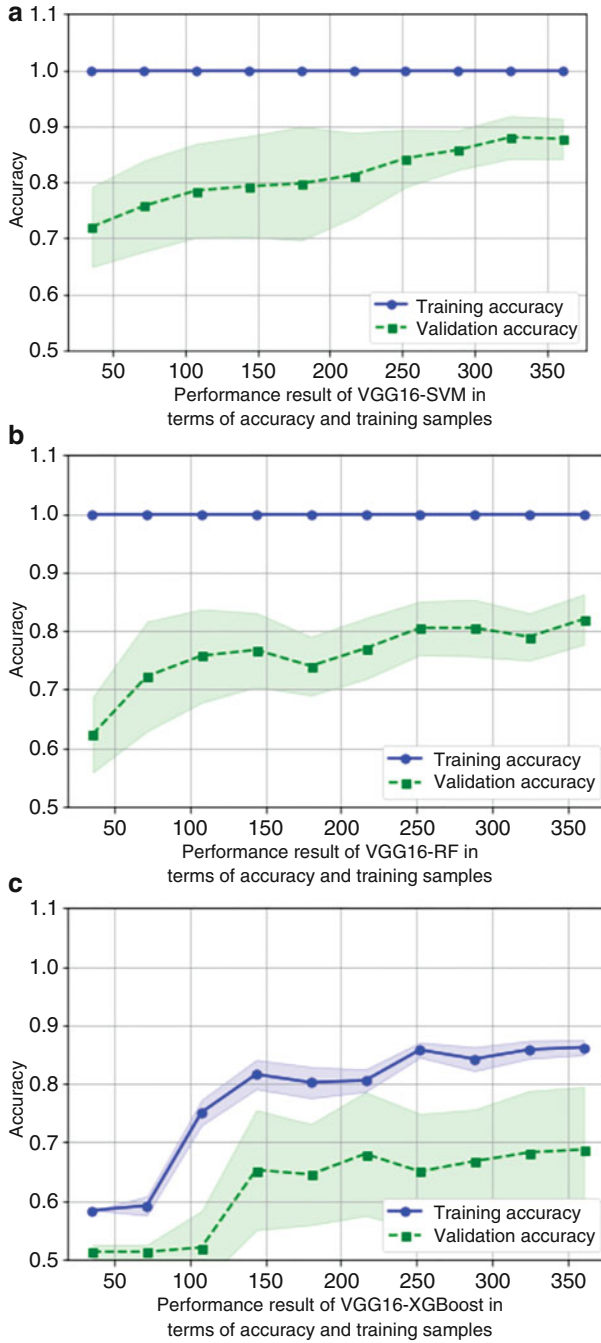


Fig. 12 Comparison of the three TL models in terms of accuracy with no. of training samples

Table 3 Experimental results of three TL models for learning curve method

	No. of samples									
	27	54	81	108	135	162	189	216	243	270
VGG16-SVM	70.66	67.33	71.66	76.00	77.33	77.00	78.00	81.33	80.66	84.66
VGG16-RF	61.00	65.33	69.33	70.00	71.00	73.33	75.66	75.66	74.66	78.33
VGG16-XGB	48.66	50.66	51.33	70.00	71.66	72.33	69.33	65.33	68.33	69.33

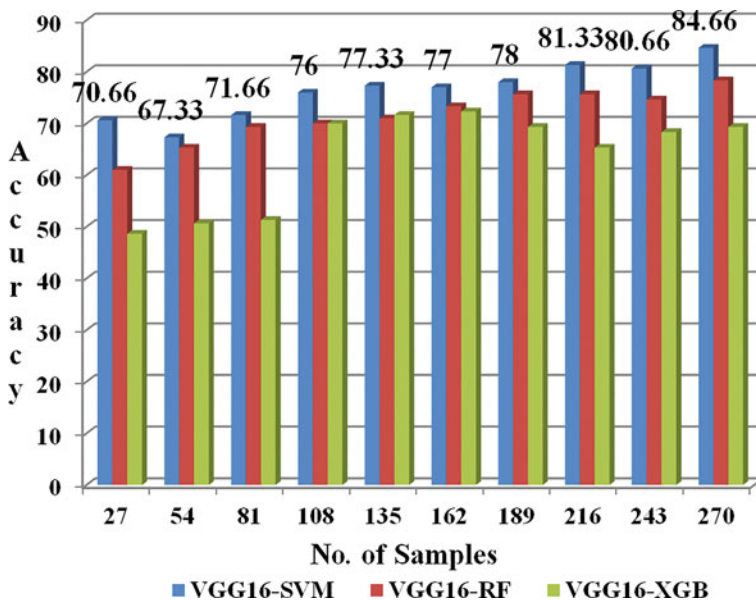


Fig. 13 Performance results of three TL models for learning curve method

4.5 Comparative Analysis

The author (G. Muscas et al. [25]) used Gradient Boost Machine in the classification of Covid-19 X-ray and obtained a result of 67.00%. A ResNet 101 CNN is employed to detect abnormalities in X-ray images by Che Azemin MZ et al. [27]. They have achieved an Accuracy of 71.9% and it is shown in the Table 4. A multi-view fusion DL model is proposed to improve the efficacy in the diagnosis of CT chest images. It has resulted in an increase in Accuracy of 4.1% than the Che Azemin MZ et al. [27] approach. S. Wang et al. [28] and H. Yasar and Ceylan [29] proposed Inception CNN and Local Binary Pattern-KNN to obtain 79.30% and 79.73% of Accuracy respectively. The author has proposed TL based VGG16-SVM to classify the Covid-19 CT Scan images which resulted in an Accuracy of 84.66%.

Table 4 Comparative analysis of proposed method with existing methods

S.No.	Reference	Methodology used	Accuracy (%)
1.	G. Muscas et al. [25]	Gradient boosting machine	67.00
2.	Che Azemin MZ [27]	ResNet 101- CNN	71.90
3.	Xiangjun Wu et al. [26]	Multi-View deep fusion model	76.00
4.	S. Wang et al. [28]	Inception CNN	79.30
5.	H. Yasar and Ceylan [29]	Local Binary Pattern-KNN 10 fold cross validation	79.73
6.	Proposed method	VGG16-SVM	84.66

4.6 Limitations

The VGG16 model suffers from a vanishing gradient problem. This can be solved through a ResNet architecture. The ResNet uses residual learning which might be used in the architecture to overcome the gradient problem. When the number of images in the dataset increases, the SVM performance will be reduced due to the requirement for high computational power. A dimensional reduction method like PCA or t-Distributed Stochastic Neighbor Embedding (t-SNE) might be used to scale down the aspect of the data for better classification with less computation.

5 Conclusion

A major virus that influences the world in the year 2020 is named as Coronavirus (COVID-19). This novel virus has out breaks the epidemic situation into pandemic one. In order to identify, prevent and control the spreading of coronavirus an effective machine learning technique is required in this situation. The work carried out helps to detect and diagnose Covid-19 automatically through medical imaging of CT-Scan images. In this study, CT-Scan images are used to classify the Covid-19 patients. Three experiments are carried out in the order SVM, RF and XGBoost to identify the proposed method. The multiple stacked convoluted neural network VGG16 in the proposed architecture is capable of extracting the complex feature from the Covid19 CT-Scan images. The extracted chronological features are fed into the SVM for classification. The SVM classifier uses a minimal sequential optimizer that iterates to find a feasible solution and it provides a better solution for spare data. The VGG16-SVM TL method obtained an accuracy of 84.66% which is greater than VGG16-RF and VGG16-XGB by 6.33% and 15.33% respectively. Thus, it can be used as a tool to identify and diagnose Covid-19 without any delay in processing. In the future, the traditional machine model can be replaced with a DL method for classifying the CT-Scan images which can obtain a better result.

References

1. Ahuja, S., Panigrahi, B. K., Dey, N., et al. (2021). Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices. *Applied Intelligence*, 51(1), 1–15. <https://doi.org/10.1007/s10489-020-01826-w>
2. Farhat, H., Sakr, G. E., & Kilany, R. (2020). Deep learning applications in pulmonary medical imaging: Recent updates and insights on COVID-19. *Machine Vision and Applications*, 31(53), 1–42. <https://doi.org/10.1007/s00138-020-01101-5>
3. Hernigou, J., Valcarenghi, J., Safar, A., et al. (2020a). Post-COVID-19 return to elective orthopaedic surgery—Is rescheduling just a reboot process? Which timing for tests? Is chest CT scan still useful? Safety of the first hundred elective cases? How to explain the “new normality health organization” to patients? *International Orthopaedics (SICOT)*, 44, 1905–1913. <https://doi.org/10.1007/s00264-020-04728-1>
4. Hernigou, J., Cornil, F., Poignard, A., El Bouchaibi, S., Mani, J., Naouri, J. F., Younes, P., & Hernigou, P. (2020b). Thoracic computerised tomography scans in one hundred eighteen orthopaedic patients during the COVID-19 pandemic: Identification of chest lesions; added values; help in managing patients; burden on the computerised tomography scan department. *International Orthopaedics*, 44, 1571–1580. <https://doi.org/10.1007/s00264-020-04651-5>
5. Perumal, V., Narayanan, V., & Rajasekar, S. J. S. (2020). Detection of COVID-19 using CXR and CT images using Transfer Learning and Haralick features. *Applied Intelligence*, 51, 1–18. <https://doi.org/10.1007/s10489-020-01831-z>
6. Pu, J., Leader, J. K., Bandos, A., et al. (2020). Automated quantification of COVID-19 severity and progression using chest CT images. *European Radiology*, 31, 436–446. <https://doi.org/10.1007/s00330-020-07156-2>
7. Sharma, S. (2020). Drawing insights from COVID-19-infected patients using CT scan images and machine learning techniques: A study on 200 patients. *Environmental Science and Pollution Research*, 27, 37155–37163. <https://doi.org/10.1007/s11356-020-10133-3>
8. Tabatabaei, S. M. H., Talari, H., Gholamrezanezhad, A., Farhood, B., Rahimi, H., Razzaghi, R., Mehri, N., & Rajebi, H. (2020). A low-dose chest CT protocol for the diagnosis of COVID-19 pneumonia: A prospective study. *Emergency Radiology*, 27, 607–615. <https://doi.org/10.1007/s10140-020-01838-6>
9. Balaji, P., Nagaraju, O., & Haritha, D. (2017). Levels of sentiment analysis and its challenges: A literature review. In *2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC)*. <https://doi.org/10.1109/ICBDACI.2017.8070879>
10. Nagaraju, O., Balaji, P., & Haritha, D. (2018). An overview on opinion mining techniques and sentiment analysis. *International Journal of Pure and Applied Mathematics*, 118(19), 61–69.
11. Penubakabalaji, P., Haritha, D., & Nagaraju, O. (2018). Feature based summarization system for e-commerce based products by using customers’ reviews. In *2018 IADS International Conference on Computing, Communications & Data Engineering (CCODE)*, pp. 1–7. <https://doi.org/10.2139/ssrn.3168342>
12. Ganesan, T., Sivakumar, S., Zeelan Basha, C. M. A. K., & Haritha, D. (2018). Classification of mining techniques in multiclass data sets using wavelets. *International Journal of Pure and Applied Mathematics*, 118(10), 217–222. <https://doi.org/10.12732/ijpam.v118i10.26>
13. Nimmagadda, S., Sivakumar, S., Kumar, N., & Haritha, D. (2020). Predicting airline crash due to birds strike using machine learning. In *2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India*, pp. 1–4. <https://doi.org/10.1109/ICSSS49621.2020.9202137>
14. Rajesh Kumar, T., Videla, L. S., SivaKumar, S., Gupta, A. G., & Haritha, D. (2020). Murmured speech recognition using Hidden Markov model. In *2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India*, pp. 1–5. <https://doi.org/10.1109/ICSSS49621.2020.9202163>

15. Sivakumar, S., Videla, L. S., Rajesh Kumar, T., Nagaraj, J., Itnal, S., & Haritha, D. (2020). Review on Word2Vec Word Embedding Neural Net. In *2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India*, pp. 282–290. <https://doi.org/10.1109/ICOSEC49089.2020.9215319>
16. Soniya, V., Swetha Sri, R., Swetha Titty, K., Ramakrishnan, R., & Sivakumar, S. (2020). Attendance automation using face recognition biometric authentication. In *IEEE 2017 International Conference on Power and Embedded Drive Control (ICPEDC), 16th–18th March 2017*, pp. 122–127. <https://doi.org/10.1109/ICPEDC.2017.8081072>
17. Vidya Sagar, S., Ragav Kumar, G., Xavier, L. X. T., Sivakumar, S., & Durai, R. B. (2020). SISFAT: Smart Irrigation System With Flood Avoidance Technique. In *IEEE 2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM), 23th–24th March 2017*, pp. 28–33. <https://doi.org/10.1109/ICONSTEM.2017.8261252>
18. Sivakumar, S., Rajalakshmi, R., Prakash, K. B., Kanna B. R., & Karthikeyan, C. (2019, July). Virtual vision architecture for VIP in ubiquitous computing. In S. Paiva (Ed), *Technological trends in improved mobility of the visually impaired* (EAI/Springer innovations in communication and computing, pp. 145–179). Springer. https://doi.org/10.1007/978-3-030-16450-8_7
19. Videla, L. S., Rao, M. R. N., Anand, D., Vankayalapati, H. D., & Razia, S. (2019). Deformable facial fitting using active appearance model for emotion recognition. In S. Satapathy, V. Bhateja, & S. Das (Eds.), *Smart intelligent computing and applications* (Smart innovation, systems and technologies) (Vol. 104). Springer. https://doi.org/10.1007/978-981-13-1921-1_13
20. Videla, L. S., et al. (2018). Modified feature extraction using Viola Jones algorithm. *Journal of Advanced Research in Dynamical and Control Systems*, 10(3 Special Issue), 528–538.
21. Videla, L. S., & Ashok Kumar, P. M. (2020). Fatigue monitoring for drivers in advanced driver-assistance system. In S. R. Nayak & J. Mishra (Eds.), *Examining fractal image processing and analysis* (pp. 170–187). IGI Global. <https://doi.org/10.4018/978-1-7998-0066-8.ch008>
22. Shanmugan, S., & Essa, F. A. (2020). Experimental study on single slope single basin solar still using TiO₂ nanolayer for natural clean water invention. *Journal of Energy Storage*, 30, 101522. <https://doi.org/10.1016/j.est.2020.101522>
23. Essa, F. A., Elsheik, A. H., Sathyamurthy, R., Muthu Manokar, A., Kandeal, A. W., Shanmugan, S., Kabeel, A. E., Sharshir, S. W., & HiteshPanchal, M. M. (2020). Younes, extracting water content from the ambient air in a double-slope half-cylindrical basin solar still using silica gel under Egyptian conditions. *Sustainable Energy Technologies and Assessments*, 39, 100712. <https://doi.org/10.1016/j.seta.2020.100712>
24. Panchal, H., Mevada, D., Sadasivuni, K. K., Essa, F. A., Shanmugan, S., & Khalid, M. (2020). Experimental and water quality analysis of solar stills with vertical and inclined fins. *Ground-water for Sustainable Development*, 11, 100410. <https://doi.org/10.1016/j.gsd.2020.100410>
25. Muscas, G., Matteuzzi, T., Becattini, E., et al. (2020). Development of machine learning models to prognosticate chronic shunt-dependent hydrocephalus after aneurysmal subarachnoid hemorrhage. *Acta Neurochirurgica*, 162, 3093–3105. <https://doi.org/10.1007/s00701-020-04484-6>
26. Wu, X., Hui, H., Niu, M., Liang, L., Wang, L., He, B., Yang, X., Li, L., Li, H., Tian, J., & Zha, Y. (2020). Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: A multicentre study. *European Journal of Radiology*, 128, 109041, ISSN 0720-048X. <https://doi.org/10.1016/j.ejrad.2020.109041>
27. Che Azemin, M. Z., Hassan, R., Mohd Tamrin, M. I., & Md Ali, M. A. (2020, August 18). COVID-19 deep learning prediction model using publicly available radiologist-adjudicated chest X-ray images as training data: Preliminary findings. *International Journal of Biomedical Imaging*, 2020, 8828855. <https://doi.org/10.1155/2020/8828855>. PMID: 32849861; PMCID: PMC7439162.
28. Wang, S., Kang, B., Ma, J., Zeng, X., Xiao, M., Guo, J., Cai, M., Yang, J., Li, Y., Meng, X., & Xu, B. (2021). A deep learning algorithm using CT images to screen for Corona virus disease (COVID-19). *European Radiology*, 31(8), 6096–6104. <https://doi.org/10.1007/s00330-021-07715-1>

29. Yasar, H., & Ceylan, M. (2021). A novel comparative study for detection of Covid-19 on CT lung images using texture analysis, machine learning, and deep learning methods. *Multimedia Tools and Applications*, 80, 5423–5447. <https://doi.org/10.1007/s11042-020-09894-3>
30. World Health Organization. (2022, August 23). *Coronavirus (COVID-19) dashboard*. <https://covid19.who.int/>
31. Transfer Learning. (2020, October 18). *Source code for proposed VGG16-SVM*. <https://github.com/SIVAKUMAR-SOUBRAYLU/VGG16-SVM-RF-XGB>
32. Loey, M., Smarandache, F., & Khalifa, N. E. M. (2020). Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning. *Symmetry*, 12(4), 651. <https://doi.org/10.3390/sym12040651>
33. Shah, V., Keniya, R., Shridharani, A., Punjabi, M., Shah, J., & Mehendale, N. (2021). Diagnosis of COVID-19 using CT scan images and deep learning techniques. *Emergency Radiology*, 28(3), 497–505. <https://doi.org/10.1007/s10140-020-01886-y>

Blockchain-Based Medical Report Management and Distribution System



Subham Kumar Sahoo, Sambit Kumar Mishra, and Abhishek Guru

Abstract Generally, the Hospital operations contain loads of scientific reviews which can be a crucial part of operations. As a result of integrating pathology and other testing labs within the medical center, hospitals today have improved their business operations while also achieving greener and faster diagnoses. Many different strategies are used in hospital operations, from patient admission and control to health center cost management. This will raise operational complexity and make it more challenging to manage, especially when combined with newly introduced offerings like pathology and pharmaceutical control. In order to overcome this issue, we employ the Hyperledger notion and a blockchain era to retain the data of each individual transaction with 100% authenticity. Instead of using a centralized server, all transactions are encrypted and kept as blocks, which are then used to authenticate within a network of computers. Additionally, we employ the hyper ledger concept to associate and store all associated scientific files for each transaction with a date stamp. This makes it possible to confirm the legitimacy of each document and identify any changes made by someone else. This consultation defines that affected person's clinical record is personal and every affected person has his very own privacy. To guard the reviews from hackers or enemies, who will make changes on clinical reviews and additionally saving the statistics without lacking any content material which performs an important position to shape a life. To study reviews, we are using a block chain method which splits the information into modules. Using this method hackers or enemies can't get the right information.

“To bring forward a secure, safe, efficient, and legitimate medical report management system” is the primary goal of this project.

Keywords Blockchain · Electronic medical record · Cloud computing · Healthcare system · Encryption algorithms

S. K. Sahoo · S. K. Mishra (✉)
SRM University-AP, Amaravati, Andhra Pradesh, India

A. Guru
Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India

1 Introduction

The Electronic Medical Record (EMR) is straightforward but extremely attentive to the innate characteristics of locating and managing human offerings, which should be shared and distributed among friends as frequently as possible. Examples of friends include pharmaceutical suppliers, insurance companies, drug stores, analysts, and patients' families, among others. This is a significant test of how well to preserve a patient's medical records. The process of a patient's therapy is just complicated by the storage and sharing of information across various components and the retaining of access to manipulation through several assents. A person who is ill, such as someone with HIV or malignant growth, wants to keep detailed records of the treatment process and post-treatment recovery and follow-up. Approaching a complete record is probably necessary for his or her therapy; for instance, understanding the transmitted radiation sections or studying facility effects is necessary for planning the treatment. Additionally, a patient may visit multiple healthcare facilities for a consultation or may be transferred from one medical facility to another. According to the law, a person who has been harmed is granted immediate control over his well-being information and may make a recommendation and restrictions on who may access it. On the rare chance that a patient wishes to share his medical records for exam purposes or exchange them from one medical facility to the next, he may be needed to sign a consent that specifies the information that can be provided, the beneficiary's records, and the time during which the records can be seen by the beneficiary. This is definitely very difficult to accomplish, especially if the affected person is traveling to a different city, region, or country and may not be aware of his parents' decision-making process or the hospital where he will receive treatment for a while beforehand.

The traditional medical record systems must follow a convoluted administrative process for the processing of data that should be able to protect and maintain the privacy of patients, which results in a significant loss of human resources. Such an architecture is ineffective for the sharing of medical records. Recently, the management and sharing of medical data have been made secure using the blockchain approach [4]. The blockchain networks' cryptographic properties ensure the patient's privacy. Data incorruptibility and integrity guard against tampering with medical data. The blockchain can be thought of as a distributed database that saves information in every network node to prevent stoppage. Thus, it offers more uniformity, attack resistance, and stability.

2 Literature Survey

Literature survey help us to find how to maintain patient control while maintaining public standards for electronic medical records, how to block the reports regarding health. In this section, a comparison has also been mentioned about owning the medical records.

The MedRec system, a distributed blockchain-based system for managing medical information, was presented by the authors. There are three different kinds of Ethereum-based deals for linking patient health records and granting other parties access to the information [5]. To facilitate the safe sharing of medical data, Yang and Yang [6] additionally proposed an attribute-based authentication method with a blockchain approach for the MedRec system, by using sign encryption. A cloud-based consortium blockchain-based electronic medical record that offers data integrity, data confidentiality, storage flexibility, and well authentication protocols is described. Through the use of smart contracts, each step of moving electronic medical records to the data center is included as a transactional event in a cooperative Ethereum blockchain [3, 7].

A blockchain-based healthcare platform [8] with a smart contract is suggested to protect the management of electronic medical records. This method offers patients a thorough, unchangeable log and simple access to personal medical data among several departments of the hospital. The authors suggested a blockchain-based approach [9] for managing health information that guarantees trust, responsibility, and transparency. A framework has been created with cancer sufferers keeping in mind. The framework's prototype provides confidentiality, and privacy, and speeds up the process of sharing medical data. The suggested system is made up of nodes, multiple Application Programming Interfaces (APIs), and managed service databases for storing medical data. The ability to evaluate the number of users or blockchain miners who are gaining access to the information contained in the blockchain system is provided by the membership service [9]. The goal of MedChain (proposed by authors [10]) is to make the existing systems better, which is built on a blockchain. Because it maintains patient confidentiality while enabling accessible, secure, and efficient access to medical records by patients, healthcare professionals, and other third parties. The authors introduce an attribute-based authentication protocol with multiple parties to ensure the reliability of EHRs embedded in the blockchain. In this scheme, a patient attests to the validity of a message based on an attribute while withholding all other information except for the proof that he has done so [11]. The authors suggested a healthcare framework that incorporates blockchain and scalable data lakes as well as a patient-centric data return methodology. Additionally, they present a few crucial architectural elements for this connection [12].

To preserve patient emergency essential medical information while the patient moves from one medical facility to some other, the authors have developed a blockchain-based approach, leaving a continuous patient footprint as a safe and scalable data source [13]. The researchers suggested a hybrid architecture that makes use of both edge nodes and blockchain to simplify user access to EHR data. A blockchain-based controller that is part of the architecture controls identification and access control regulations and acts as a tamper-proof log of accessing activities [14]. By combining user-generated acceptable use guidelines with smart contracts, the authors provide a blockchain technology framework for the development of secure management of private information in a health information exchange [15]. For improved data administration, the author suggests many workflows for the

healthcare sector or ecosystem using blockchain technology [22, 23]. The authors' proposed framework [31] includes several patient users who want their medical records to be maintained on the blockchain to ensure confidentiality and safety in the healthcare sector. A new block is added to the current blockchain architecture each time a new patient is added. The blocks form a distributed network that is connected to one another. In the current blockchain, each block has a timestamp that is validated using cryptographic techniques and includes the hashing value of the preceding block.

According to the author, decentralization has the potential to increase a little bit the security of healthcare. So, they are suggesting a healthcare data management system that uses blockchain technology since it encourages integrity and honesty. By securing patient information using various cryptographic techniques, confidentiality is guaranteed. The authors' suggested technique includes many different entities and functions [32].

2.1 How to Maintain Patient Control While Maintaining Public Standards for Electronic Medical Records

An affected person's scientific information are typically fragmented throughout more than one remedy web sites, posing an impediment to scientific care, research, and public fitness efforts.

- (a) Electronic scientific information and the net offer a technical infrastructure on which to construct longitudinal scientific information that may be included throughout web sites of care. Choices approximately the shape and possession of those information could have profound effect at the accessibility and privateness of affected person data. Already, alarming tendencies are obvious as proprietary on-line scientific file structures are advanced and deployed. The generation promising to unify the presently disparate portions of an affected person's scientific file can also additionally absolutely threaten the accessibility of the data and compromise patients' privateness [1].
- (b) In this text we recommend doctrines and 6 appropriate traits to manual the improvement of on-line scientific file structures. We describe how such structures might be advanced and used clinically.

2.2 Blocking of Health Information Report

When people and organizations purposefully prevent the interchange or use of electronic health information, this is known as health information blocking. ONC's research analyses the amount of information blocking as it is now understood, offers criteria for detecting it and separating it from other interoperability barriers,

and outlines actions that the federal government and the private sector can do to discourage this activity.

If the Office of the National Coordinator for Health Information Technology (ONC) wants to ensure that Certified Electronic Health Record Technology (CEHRT) provides cost to qualified hospitals, eligible vendors, and taxpayers, it is urged that it apply its certification application carefully. Only those products that genuinely meet modern-day major use application standards and that do not obstruct the transmission of health information should be certified by ONC. ONC needs to take steps to decertify merchandise that proactively blocks the sharing of facts due to the fact the one's practices frustrate congressional intent, de-value taxpayer investments in CEHRT, and make CEHRT much less precious and extra burdensome for eligible hospitals and eligible vendors to apply. The settlement requests an in-depth record from ONC no later than 90 days after enactment of this act concerning the quantity of the facts blockading problem, consisting of an estimate of the wide variety of carriers or eligible hospitals or vendors who block facts. This certain record needs to additionally consist of a complete method on the way to deal with the facts blockading issue [2].

3 Existing System

It is based on a trusted party in comparison to the current cloud-based fully digital scientific file machine and can experience delay or lag while accessing data from across the internet. Patient information could be compromised if merged with that of other users, and the cloud-based, entirely digital medical record system does not provide patients complete control. Cloud storage typically uses Attribute Based Encryption (ABE) and Key Aggregate Cryptosystems (KAC) [16] for information sharing. The scientific blockchain machine is tamper-evident and offers privacy safety and reliable storage when compared to the ABE and KAC.

Disadvantages

- High complexity
- Low Computation
- Requires skilled persons

Proposed System

Our proposed system framework for managing medical records applies the Ethereum-based blockchain in place of the conventional centralized databases to assure data security. By implementing smart contracts, the blockchain networks' nodes can store medical records. Additionally intended to reduce human resource waste and expedite the medical process are the automatic smart contracts for the administration process.

As an example of how this results in the secure, environmentally friendly, and reliable control of the entire system, we have employed blockchain technology

to keep track of every transaction with 100% authenticity using the hyper ledger concept. When a blockchain device is successfully installed, it will automatically run scientific information files continuously and lower community backup without the need for catastrophe recovery costs. Sharing medical information can increase costs while lowering the cost of transmitting scientific data.

Advantages

- Good Accuracy Rate
- Low Complexity
- Advanced Computation
- No requirement for professionals

4 Integration of Blockchain and Cloud

The way technological elements come together to create a cloud, where resources are aggregated over virtualization techniques and accessible across a network, is known as cloud architecture.

Cloud includes three basic services:

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

4.1 SaaS

Software as a Service is a methodology for providing services and applications through the internet. Here, the vendor is responsible for performing hardware and software maintenance. We don't need to install the software on our computer or other hardware. The price of maintaining the software and hardware was therefore set aside. As a part of SaaS model, clients are licensed with software program. Usually, whenever there is a demand based on that only licenses are made or on a pay-as-you-go arrangement. It utilizes a common model (one software is used by multiple clients). It is accessible anytime anywhere and time is also reduced as we can use the applications directly from browser. This type of system can be found in Gmail, Microsoft Office's 365, Drop box, Google Drive, Salesforce, Cisco Webex. In comparison to IaaS and PaaS, we as a customer or end user get less control in SaaS.

4.2 PaaS

Platform-as-a-Service is the most challenging of the three cloud computing layers. Although PaaS and SaaS have some similarities, PaaS is a platform for developing software that is given over the Internet as opposed to providing software as a service. Typically, this paradigm is used by developers to offer a platform and environment (i.e., Runtime Environment), which in turn enables the developers to create online applications and services. PaaS provides the instruments required to create and deploy applications. Web browsers are used to access PaaS services, which are hosted in the cloud. Hence here the control is more than the SaaS model, but we don't have the control over infrastructure, network, servers, and storage. i.e., Only the User Interface will be used in our interactions; the vendor will supply the Operating System. The deployed apps in this case are under our control, as well as perhaps some configuration options for the application hosting environment. This model includes platforms like [Salesforce.com](https://www.salesforce.com), Google App Engine, Heroku.

4.3 IaaS

Infrastructure as a Service refers to a way of supplying everything like operating systems, servers, and storage. All can be accessed using IP-based connectivity as part of an on-demand service. Instead of needing to purchase software and servers entirely, customers can use an on-demand and outsourced service. IaaS, a form of cloud computing service utilized by system administrators and network architects, essentially offers infrastructure. This merely offers the servers, network, security, and operating system that will be used to install the applications. i.e., It allows users to connect to fundamental resources such as physical machines, virtual machines, virtual storages, etc. Because we have complete control over computing resources thanks to administrative access to virtual machines, IaaS offers more control than SaaS and PaaS. Popular examples of the IaaS system include AWS (EC2), IBM Cloud, Oracle Cloud infrastructure, Google Cloud and Microsoft AZURE.

Blockchain

By writing an article for Bitcoin in 2008, Nakamoto (2018) revealed the concept of blockchain technology [26, 27]. An ever-expanding collection of data blocks makes up a data structure known as a blockchain. It is a continuously expanding linked list-like structure of records that consists of blocks that are connected by links and are stored using encryption. A timestamp indicating when the block was added to the blockchain, a cryptographic copy of the block before it, and transaction-related data are all included in each block of the blockchain. Since each block holds a connection to its predecessor and information about it, the preceding block cannot be altered or removed. Blockchain can avoid changing any data in any of the blocks because, once the data is recorded, it cannot be changed without also changing the data in all the blocks that come before it [18–20, 24, 25]. Clinical trial misconduct is handled

effectively by blockchain, and this prospective helps to increase the efficiency of data for the healthcare industry [28].

Some major challenges of the cloud-like [29, 30]:

1. Centralized communication models
2. Trusting a third-party cloud provider for data privacy
3. Higher latency for large-scale deployment, has been the motivation for the integration of Cloud Computing/Cloud of things and Blockchain.

Benefits like scalability, decentralization, increase in data security, and uninterrupted services by minimizing the failure risk can be achieved in cloud computing by using blockchain.

Electronic Health Records (EHRs) will be processed and kept online on cloud storage, and doctors will be able to access patient medical data via mobile devices (such as smartphones) to monitor health. With the support of decentralized data verification across all peer nodes, blockchain can also be demonstrated to be crucial in resolving security issues in the process of sharing health data [5, 17, 22].

5 Encryption/Decryption

A brief idea is presented below for data encryption and decryption.

5.1 Data Encryption

Data encryption converts information into a different form, so that only people with access to a secret key (formally referred to as a decryption key) may interpret it. Those facts which are not encrypted noted as plaintext, whereas encrypted data are typically noted as ciphertext. Encryption is currently one of the most well-known and effective data security methods used by businesses. The two primary types of data encryption are symmetric encryption and uneven encryption, also referred to as public-key encryption.

When virtual information is kept on computer systems and transmitted via the internet or other computer networks, information encryption is used to protect its confidentiality. The outdated data encryption standard (DES) has been updated with modern encryption methods, which are essential for the security of IT systems and communications. These algorithms offer confidentiality and compel crucial safety responsibilities with authentication, integrity, and non-repudiation. Integrity of a message shows that its contents have not changed since it was sent, and authentication enables origin verification. A message sender cannot cancel the message's transmission thanks to non-repudiation.

Various encryption mechanism is presented by authors [21], which can be used for cryptographic techniques. These are:

1. Broadcast
2. Identity-based
3. Attribute-based
4. Re-encryption proxy
5. Searchable symmetric

5.2 Data Decryption

Decryption is the method of remodeling facts that has been rendered unreadable via encryption returned to its unencrypted form. In decryption, the machine extracts and converts the garbled facts and transforms it to texts and pictures which are without problems comprehensible now no longer most effective with the aid of using the reader however additionally with the aid of using the machine. Decryption can be achieved manually or automatically. It will also be executed with a hard and fast of keys or passwords. One of the most motives for enforcing an encryption decryption machine is privacy.

6 Software Development Life Cycle (SDLC)

We are using the waterfall model for our project's software development cycle due to its step-by-step implementation process (Fig. 1).

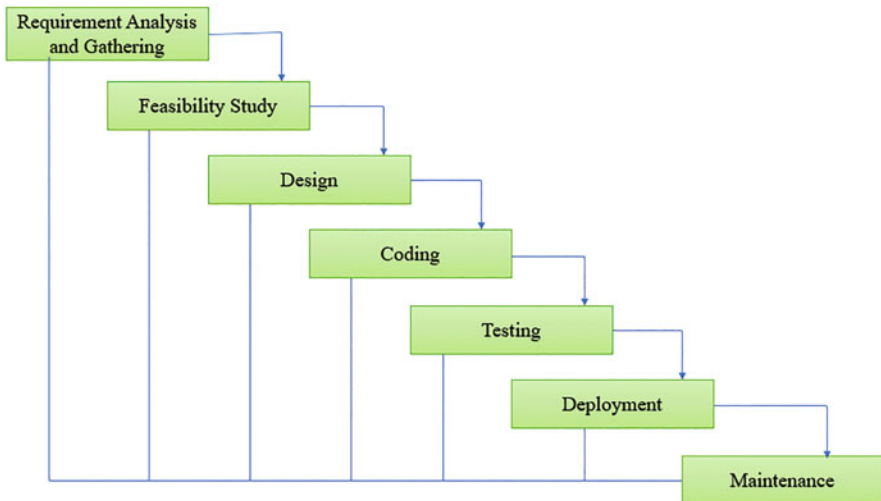


Fig. 1 SDLC diagram

- **Requirement Analysis and Gathering** – In this stage, every potential requirement for the system that will be created. The same will be assembled and outlined in a document that contains requirements and specifications. Some important questions like who will use the system, what is the need of this software, what is the future scope of this are discussed here.
- **Feasibility Study** – In the second phase, the developing organization discusses the costs and benefits of that software they are going to developed. Profit plays an important role, because the organization has a fear of loss also if the cost is too high.
- **Design** – The outlet design of the software will be started in the design phase by the architect. The SRS document that is created in this phase includes all the logical details such as database design, language that is used, etc. of the model. A prototype of the final outcome is produced in this phase.
- **Coding** – After designing the software, coding is started by the developers using any programming language. Different modules are coded by the developers, and all are arranged in an efficient manner to get the effective outcome of the product.
- **Testing** – After the completion of the coding phase, testing is performed on the code by different teams. The testing team can identify the bugs in the code. If any of the bug is identified, the code is sent back to the developers to modify the code. After modifying the errors or bugs in code, the code is again tested.
- **Deployment** – After the completion of testing phase, the product will be free from bugs. The product will be delivered to the customers and the evaluation of the product is explained for the usage. Even though if there are bugs in the product, then the product is taken under maintenance, and it is again deployed to the customers as a new version product.
- **Maintenance** – Several setbacks can occur on the client side. Patches are made available to label those problems. Additionally, some newer iterations of the product have been launched. To implement these modifications in the client environment, maintenance is performed.

7 System Requirements Specification or Analysis

The complete description and specification of the requirements that are used to generate the product that satisfies the needs of the users are specified in a document known as SRS. It lays the groundwork required to ensure that everyone involved in the project is aware of the most crucial information. An SRS outlines the tasks, competencies, and capacities required by the system as well as any potential restrictions. Needs might be functional or non-functional. Design proposals and information that are not part of the customer's needs are not provided.

All required stakeholders have given their consent, demonstrating that they have a thorough understanding of the project's requirements and that everyone agrees. The SRS serves as a sort of insurance policy that any party may turn to in case of ambiguity.

7.1 *Functional Requirements*

These are the basic requirements that the device must provide for the give-up person in particular. As stipulated in the contract, all those features must be permanently incorporated into the device. These are shown or stated inside the device’s form of entry, along with the operation that was carried out and the results anticipated. They are generally the necessities listed by the person who you will immediately observe inside the final product, as opposed to the unnecessary necessities.

7.2 *Non-useful Necessities*

These are effectively the only requirements that the device must meet to comply with the job contract. From one task to another, the ingredients may be applied in a different order or quantity. They are also referred to as non-behavioral needs.

They essentially address problems like portability, security, maintainability, reliability, ability to scale, the performance, reusability, and flexibility.

8 System Design

The specifications that are required for designing of medical blockchain that are as follows:

8.1 *System Specifications*

Operating System	Windows 10
Server-Side Script	Python 3.6
IDE	PyCharm
Library	Pandas, MySQL
Framework	Flask

8.2 *Input Design*

Input refers to the unprocessed data that an information system uses to produce output. Developers must take into account input devices like Personal Computers

(PC), Magnetic Ink Character Recognition (MICR), Optical Mark Reading (OMR), etc. during input design.

Therefore, the output quality of a system is determined by the input quality. Entry forms and screens with good design include the following characteristics:

- It should successfully accomplish a certain objective such as storing, recording, and retrieving data.
- It ensures precise and appropriate execution.
- Entries should be straightforward and simple to complete.
- Mainly includes the attention of user and integrity of the product.
- Utilizing a foundational understanding of design principles, all these goals are achieved by considering
 - What are the inputs required for the system?
 - How do consumers react to various features on displays and forms?

Objectives for Input Design

These are the goals of input design:

- To design strategies for data entry and input.
- To lower the input volume.
- The creation of source documents for data collection or work out alternative data collection techniques.
- To create user interface displays, data entry screens, input data records, etc.
- To implement validation checks and create efficient input controls.

8.3 Output Design

Output design is the most important responsibility for every system. Developers determine the vital output types and consider the output controls and report prototype layouts during output design.

Objectives of Output Design

Objectives of output design are as follows:

- To create output designs that fulfill the desired function and stop the creation of undesired output.
- Output should be designed in such a way that; it should meet the end user's requirements.
- To provide the proper amount of production.
- To create the output in the suitable format and send it to the relevant recipient.
- To deliver the output on time.

8.4 *Implementation*

Users who have access to the shared file can do so if they have the ability (i.e., the decryption key) necessary to administer policy. Owners of Personal Health Records (PHRs) upload data files, such as treatment records, patient profiles, etc., in an encrypted format to a cloud server. Owners and users of PHR are given a set of characteristics in the form of a personal decryption key by the characteristic government.

Our version supports many governments that can challenge the attributes of customers. For instance, a patient may receive keys from a private government entity, such as a hospital or insurance provider. A proxy server that is a semi-relied server is used in the data outsourcing environment that includes cloud computing to be in charge of functioning as a re-encryption challenge while any coverage is updated. Thus, the bulk of crucial computational processes and steady cryptographic procedures are uploaded to the proxy. The proxy is connected to an X.509 certificate that was generated by a reputable Certifying Authority (CA). The usage of certificates for authentication with various device entities. As a result, the proxy connects in the best way with the entities holding the valid certificates that were predetermined in its configuration device.

9 **Modules**

A quick description has been presented for the modules that have been designed for this management system. Those are Admin, Receptionists, Doctor, and Patient.

Admin

Admin can login with valid credentials; admin can add doctors, add receptionists, view doctors, view receptionists and view patients.

Receptionists

Receptionists can log in with provided email id and password, they can add patients, update their details, and view patient details and generate bills.

Doctor

Doctors can login with provided email id and password. They can view patients, view their reports, prescribe tests and medicines.

Patient

Patients can login through their credentials to view their profile and reports.

10 UML Diagrams

A standardized widespread modeling terminology known as UML is used in the field of object-oriented software engineering.

The objective is for UML to establish itself as a commonplace language for initiating an object-oriented device software design. In its current form, UML is built up primarily of two elements: a Meta-version and a notation. A few different types of techniques or methods will be added to or connected to UML in the future. The Unified Modeling Language is a popular interface for non-software systems and business analytics as well as for itemizing, visualizing, building, and writing down the artifacts of software systems.

The UML represents a group of quality engineering procedures that demonstrated a success within the modeling of huge and complicated systems.

The UML is a completely critical section of growing objects-oriented software programs and the software program improvement method. The UML primarily employs visual notations to specify the organization of software program designs.

Goals

The Primary goals in the design of the UML are as follows:

1. Present customers a ready-to-use, language of expressive visible modeling with the intention to expand and interchangeably notable models.
2. To raise middle standards, this offers tools for extensibility and specialization.
3. Be impartial to unique programming languages and development processes.
4. Establish a strong framework for modelling language skills.
5. Encourage the roar of the Object-Oriented appliances market.
6. Support collaborations, frameworks, styles, and components as well as standards for higher degree improvement.
7. Integrate quality practices.

10.1 Use Case Diagram

A use case diagram is a specific kind of behavioral diagram that is created using the Unified Modeling Language (UML) and is specified by and derived from a use-case study.

Use cases serve as a representation of high-level functionality and how users will interact with the system. A use case shows how a system, component, package, or class has a certain functionality. A circle with the use case name written inside, serves as a symbol for it. An individual that engages with the system is the actor and a user is the best example of this. An actor is a thing that starts a use case from outside its boundaries. Any component that has the potential to interact with the use case qualifies. In the system, one actor may be connected to several use cases (Fig. 2).

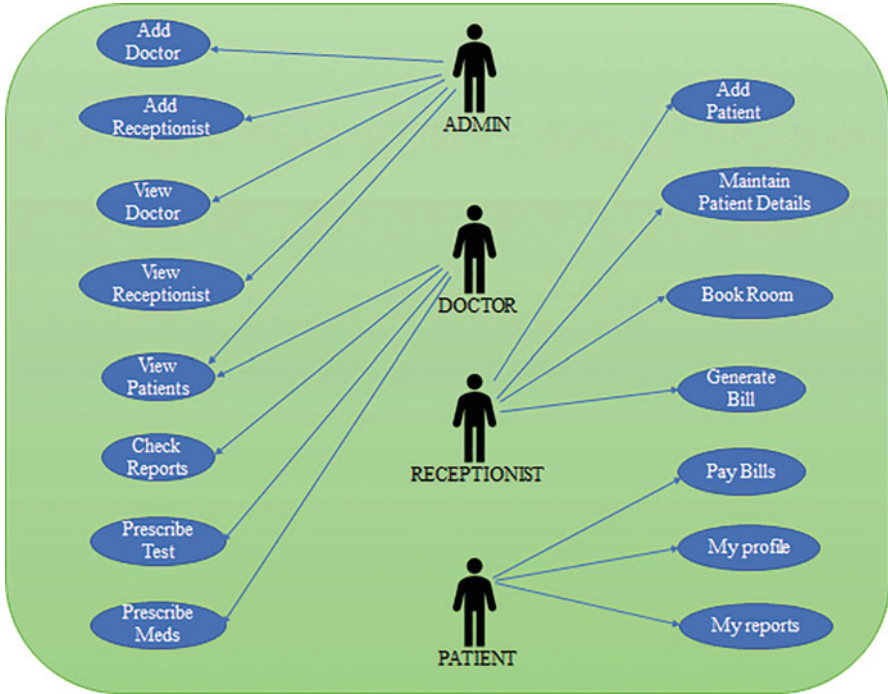


Fig. 2 UML diagram

There are four actors with the names admin, doctor, receptionist, and patient in the use case diagram up top. A medical report management system’s specialized capability is represented by a total of 15 use cases. Every actor engages with a certain use case. A system administrator can look up doctors, receptionists, and patients. Even though there are still additional use cases in the system, this actor is limited to only these interactions with it. Each actor need not engage with each of the use cases, but it is conceivable.

The second actor, a doctor, can engage with the system by looking up patients and their reports in it, as well as by recommending tests and medications for the patients. The third actor who can deal with new patients, reserve rooms for them, and keep track of every detail is the receptionist. The fourth actor, the patient, interacts with his own statistics and profile. The interactions of each actor together represent the medical report management system.

10.2 Class Diagram

The system structure is mapped by the class diagram using classes, attributes, relations, and activities among various objects, is one of the UML diagram types

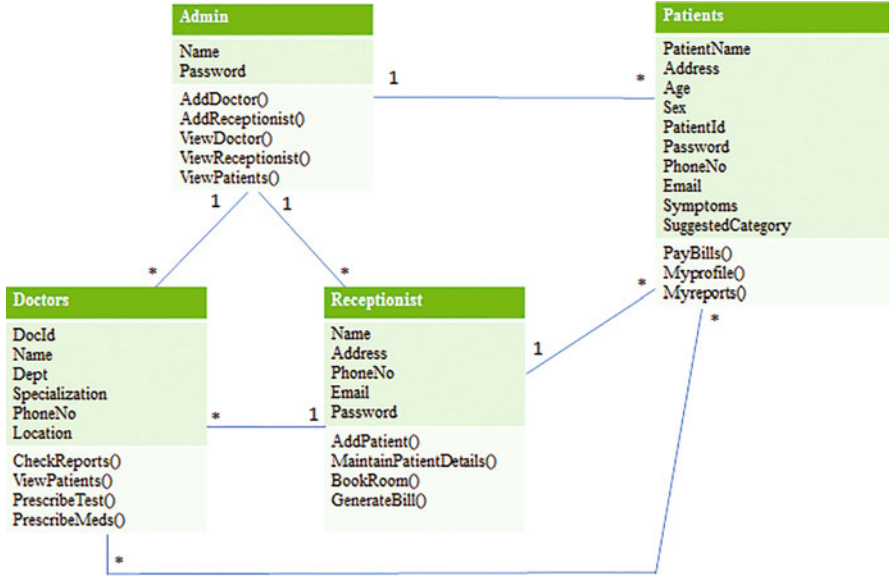


Fig. 3 Class diagram

used to explain static diagrams. Different classes are represented in a class diagram, and each class is broken down into three sections: the first section of the class diagram contains the class name, which is the name of the class or entity that took part in the activity; the second section of the class diagram contains the class attributes; the third section of the class diagram contains the class operations, and relationships show the relationship between two classes. Here four classes (Admin, Doctor, Receptionist, Patient) have been defined with some of their attributes, operations, and relationship between the classes has also been drawn (Fig. 3).

10.3 ER Diagram

An ER diagram defines the relation between the entities in a system. To build relational databases, ER diagrams are used in the fields of software engineering, business information systems, tutoring, and experimentation. They are also known as ERDs (Entity Relationship Diagrams) or ER (Entity Relationship) Models, and they make use of a predetermined group of symbols to depict how entities, relationships, and attributes are interdependent. Some of these symbols include rectangles, diamonds, ovals, and connecting lines. They comprise verbs for relationships and nouns for entities, much like in grammatical patterns.

Below is the ER diagram for Medical Report Management where there are four entities namely Patient, Doctor, Receptionist, and Admin. Each entity contains

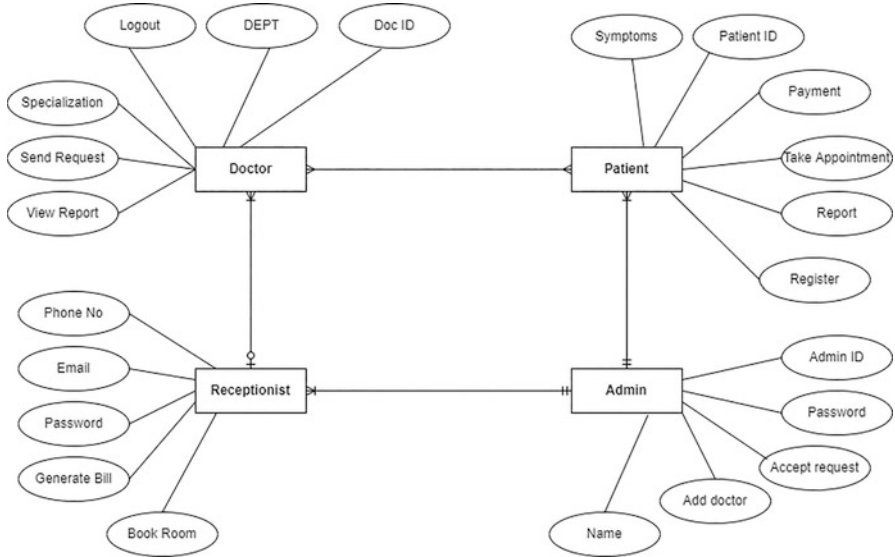


Fig. 4 ER diagram

multiple attributes and there exists a relationship between two entities. There exists a M: N relationship between Patient and Doctor, that means multiple patients can take medication from multiple Doctors. There exists 1:M relationship between Admin and Patient that indicates one admin will have complete details of the Patient. There exists 1:M relationship between Admin and Receptionist that indicates one admin will have all the control over all the receptionists (Fig. 4).

10.4 Deployment Diagram

A physical system that completes the test execution by executing the software design and showing how the software interacts with the hardware. One of the types of UML diagrams used to specify the hardware requirements for a given product to run the software is the UML deployment diagram. Node, component, artifact, and interface notation are used to design the deployment diagram. Here Admin, Receptionist, Doctor, Patient are connected to a database through a system (Fig. 5).

10.5 Collaboration Diagram

The collaboration diagram depicts the relationships between the system’s objects. Both the sequence diagram and the collaboration diagram present the same facts in

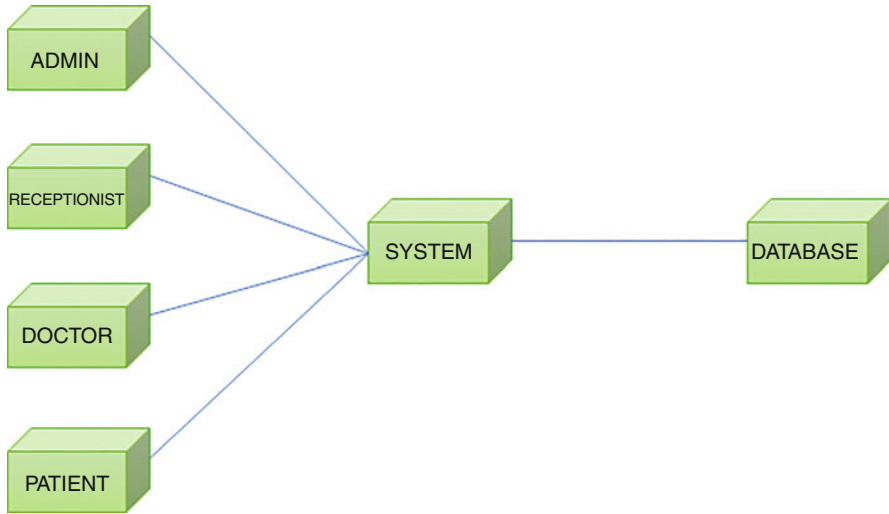


Fig. 5 Deployment diagram

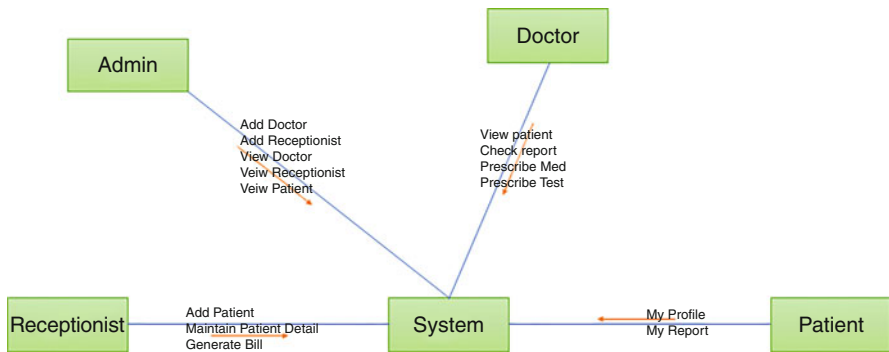


Fig. 6 Collaboration diagram

dissimilar ways. Because it is based on object-oriented programming, this diagram displays the layout of the object that resides in the system instead of the flow of messages. Various aspects make up an object. The numerous objects in the system are interconnected. The collaboration diagram, also known as a communication diagram, displays the architecture of the object within the system. Notations like objects, actors, links, and messages are used in collaboration diagrams. When it is crucial to show the interaction between the objects, collaborations are used. Collaboration diagrams are the finest tool for use case analysis (Fig. 6).

10.6 DFD Diagram

A Data Flow Diagram (DFD) is a common way to depict how information moves through a device. A neat DFD can graphically depict a staggering amount of gadget requirements. It can be manually operated, automatically operated, or a combination of the two. It describes how facts enter and exit the device, what modifies the facts, and where facts can be stored. A DFD is used to reveal a device’s range and limitations. It can be utilized as a connecting tool between a structural analyst and anyone who performs a function inside the apparatus that serves as the starting point for system remodeling.

Each entity must go through a login module here. Patients have to login and register themselves to check their details like reports, tests as well as for their appointments and bookings of rooms. Receptionists update and maintain all details regarding the patient, and information regarding allocated doctors. Doctors can check the reports of their patients, prescribe tests and medicine for patients. They can also check for appointments through their login credentials (Fig. 7).

Test Cases

Input	Output	Result
Input text files	File upload or not	Success

Testcases Modelbuilding

S.NO	Test cases	I/O	Expected O/T	Actual O/T	P/F
1	Read data	File data	Data read successfully	Data read success	P
2	Performing encryption on file data	Encryption must perform on file data	Encryption should be performed on file data	Encryption successfully completed	P
3	Generating key pair	Key must generate	Key will generate	Key generated successfully	P
4	Cipher text	File data encrypted data will decrypt	Data should be decrypt	Data decrypted successfully	P

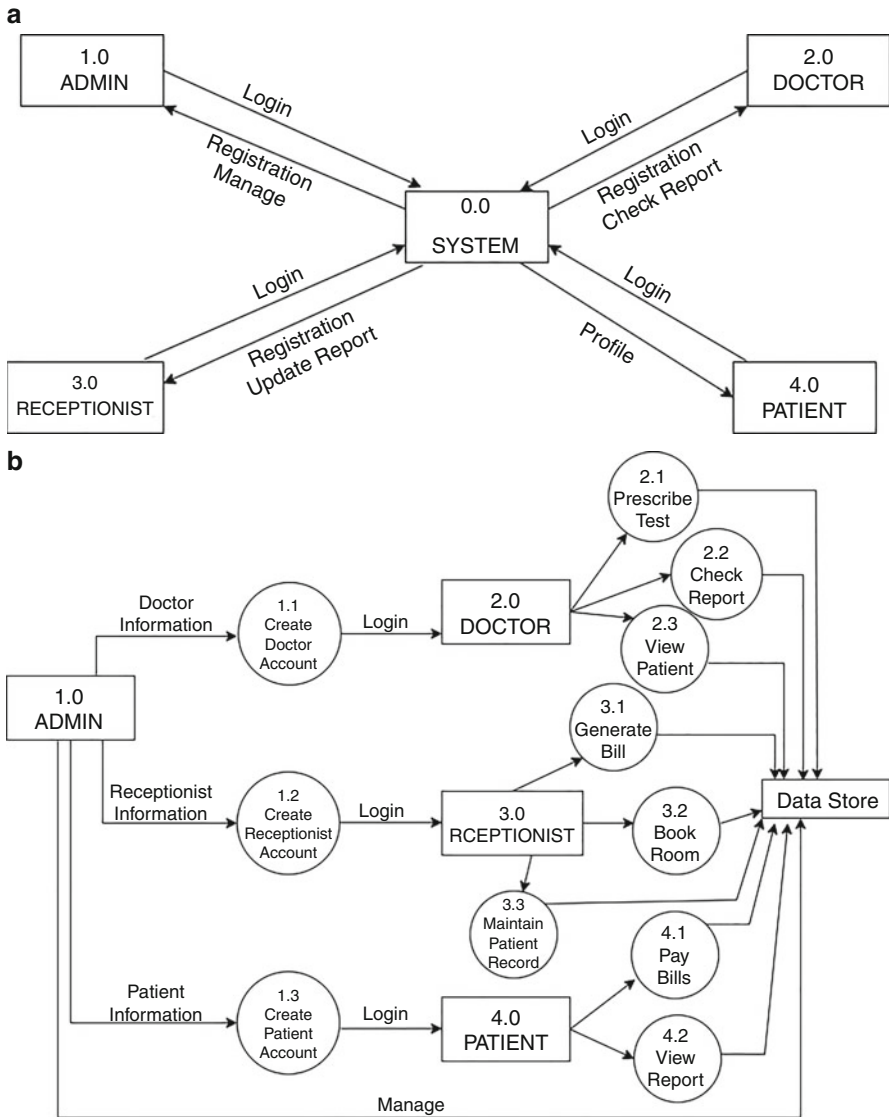


Fig. 7 (a) DFD level 0 diagram. (b) DFD level 1 diagram

11 Conclusions and Future Scope

Once a blockchain machine has been successfully built, it will automatically run reports and clinical statistics continuously and back up the entire community without the need for expensive disaster restoration. Sharing medical data can result in unnecessary costs and reduce the value of clinical data exchanges.

We will conduct in-depth tests to evaluate the cloud-based proxy with a bigger data volume and more expansive access controls in a genuine cloud environment.

References

1. Mandl, K. D., Markwell, D., MacDonald, R., Szolovits, P., & Kohane, I. S. (2001). Public standards and patients' control: How to keep electronic medical records accessible but private Medical information: Access and privacy Doctrines for developing electronic medical records Desirable characteristics of electronic medical records Challenges and limitations for electronic medical records Conclusions Commentary: Open approaches to electronic patient records Commentary: A patient's viewpoint. *BMJ*, *322*(7281), 283–287.
2. The Office of the National Coordinator for Health Information Technology. (2015). *Report on health information blocking* (Technical Report). U.S. Department of HHS.
3. Wang, H. L., Chu, S. I., Yan, J. H., Huang, Y. J., Fang, I. Y., Pan, S. Y., & Shen, T. T. (2020). Blockchain-based medical record management with biofeedback information. In *Smart biofeedback-perspectives and applications*. IntechOpen.
4. Jin, H., Luo, Y., Li, P., & Mathew, J. (2019). A review of secure and privacy-preserving medical data sharing. *IEEE Access*, *7*, 61656–61669.
5. Azaria, A., Ekblaw, A., Vieira, T., & Lippman, A. (2016). MedRec: using blockchain for medical data access and permission management. In *2016 2nd international conference on Open and Big Data (OBD)*.
6. Yang, H., & Yang, B. (2017, November). A blockchain-based approach to the secure sharing of healthcare data. In *Proceedings of the Norwegian information security conference* (pp. 100–111). Nisk J.
7. Exceline, C. E., & Nagarajan, S. (2022). Flexible access control mechanism for cloud stored EHR using consortium blockchain. *International Journal of Systems Assurance Engineering and Management*, 1–16.
8. Hang, L., Choi, E., & Kim, D. H. (2019). A novel EMR integrity management based on a medical blockchain platform in hospital. *Electronics*, *8*(4), 467.
9. Dubovitskaya, A., Xu, Z., Ryu, S., Schumacher, M., & Wang, F. (2017). Secure and trustable electronic medical records sharing using blockchain. In *AMIA annual symposium proceedings* (Vol. 2017, p. 650). American Medical Informatics Association.
10. Daraghmi, E. Y., Daraghmi, Y. A., & Yuan, S. M. (2019). MedChain: A design of blockchain-based system for medical records access and permissions management. *IEEE Access*, *7*, 164595–164613.
11. Guo, R., Shi, H., Zhao, Q., & Zheng, D. (2018). Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems. *IEEE Access*, *6*, 11676–11686.
12. Steria, S. (2021). A blockchain-based healthcare platform for secure personalised data sharing. *Public Health and Informatics: Proceedings of MIE*, *281*, 208.
13. Hasavari, S., & Song, Y. T. (2019). A secure and scalable data source for emergency medical care using blockchain technology. In *2019 IEEE 17th international conference on Software Engineering Research, Management and Applications (SERA)*, pp. 71–75.
14. Guo, H., Li, W., Nejad, M., & Shen, C. C. (2019). Access control for electronic health records with hybrid blockchain-edge architecture. In *2019 IEEE international conference on blockchain (Blockchain)*, pp. 44–51.
15. Amofa, S., Sifah, E. B., Kwame, O. B., Abla, S., Xia, Q., Gee, J. C., & Gao, J. (2018). A blockchain-based architecture framework for secure sharing of personal health data. In *2018 IEEE 20th international conference on e-Health Networking, Applications and Services (Healthcom)* (pp. 1–6). IEEE.

16. Sikhar, P., Shrivastava, Y., & Mukhopadhyay, D. (2016). Provably secure key-aggregate cryptosystems with broadcast aggregate keys for online data sharing on the cloud. *IEEE Transactions on Computers*, 66(5), 891–904.
17. Zhang, P., White, J., Schmidt, D. C., Lenz, G., & Rosenbloom, S. T. (2018). FHIRChain: Applying blockchain to securely and scalably share clinical data. *Computational and Structural Biotechnology Journal*, 16, 267–278.
18. Krichen, M., Ammi, M., Mihoub, A., & Almutiq, M. (2022). Blockchain for modern applications: A survey. *Sensors*, 22, 5274.
19. Siyal, A. A., Junejo, A. Z., Zawish, M., Ahmed, K., Khalil, A., & Soursou, G. (2019). Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives. *Cryptography*, 3(1), 3.
20. Ng, W. Y., Tan, T. E., Movva, P. V., Fang, A. H. S., Yeo, K. K., Ho, D., & Ting, D. S. W. (2021). Blockchain applications in health care for COVID-19 and beyond: A systematic review. *The Lancet Digital Health*, 3(12), e819–e829.
21. Kirupanithi, D. N., & Antoniodoss, A. (2020). Analyzing the cost efficiency using attribute based encryption on medical blockchain platform. *International Journal of Electrical Engineering and Technology*, 11(3), 394–407.
22. Roehrs, A., da Costa, C. A., da Rosa Righi, R., da Silva, V. F., Goldim, J. R., & Schmidt, D. C. (2019). Analyzing the performance of a blockchain-based personal health record implementation. *Journal of Biomedical Informatics*, 92, 103140.
23. Khatoon, A. (2020). A blockchain-based smart contract system for healthcare management. *Electronics*, 9.1, 94.
24. Agbo, C. C., Mahmoud, Q. H., & Eklund, J. M. (2019, April). Blockchain technology in healthcare: A systematic review. *Healthcare*, 7(2), 56. MDPI.
25. Hölbl, M., Kompara, M., Kamišalić, A., & Nemeč Zlatolas, L. (2018). A systematic review of the use of blockchain in healthcare. *Symmetry*, 10(10), 470.
26. Khan, F. A., Asif, M., Ahmad, A., Alharbi, M., & Aljuaid, H. (2020). Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development. *Sustainable Cities and Society*, 55, 102018.
27. Hasselgren, A., Kralevska, K., Gligoroski, D., Pedersen, S. A., & Faxvaag, A. (2020). Blockchain in healthcare and health sciences—A scoping review. *International Journal of Medical Informatics*, 134, 104040.
28. Haleem, A., Javaid, M., Singh, R. P., Suman, R., & Rab, S. (2021). Blockchain technology applications in healthcare: An overview. *International Journal of Intelligent Networks*, 2, 130–139.
29. Mishra, S. K., Sahoo, B., & Parida, P. P. (2020). Load balancing in cloud computing: A big picture. *Journal of King Saud University-Computer and Information Sciences*, 32(2), 149–158.
30. Mishra, S. K., Puthal, D., Sahoo, B., Jena, S. K., & Obaidat, M. S. (2018). An adaptive task allocation technique for green cloud computing. *The Journal of Supercomputing*, 74(1), 370–385.
31. Esposito, C., De Santis, A., Tortora, G., Chang, H., & Choo, K. K. R. (2018). Blockchain: A panacea for healthcare cloud-based data security and privacy? *IEEE Cloud Computing*, 5(1), 31–37.
32. Al Omar, A., Rahman, M. S., Basu, A., & Kiyomoto, S. (2017, December). Medibchain: A blockchain based privacy preserving platform for healthcare data. In *International conference on security, privacy and anonymity in computation, communication and storage* (pp. 534–543). Springer.

Design of 3-D Pipe Routing for Internet of Things Networks Using Genetic Algorithm



Vivechana Maan, Aruna Malik, Samayveer Singh, and Deepak Sharma

Abstract The problem of routing paths has been studied systematically for several years. The problem is quite deterministic when there are only one start and ending point and there are many well-known path-finding algorithms like Dijkstra's, A*, Maze algorithm. But for these algorithms both efficiency and correctness fall terribly when applied on more than two connection points (pair of connection points). To overcome these problems, a genetic algorithm (GA) which is an evolutionary-based biologically inspired technique is used. It tries to search global minimal from the overall solution by slowly improving over the previous solution. GA has already proved useful to solve many complex engineering problems. It is mostly used for searching and optimization related problems. The existing methods for a similar path routing problem for internet of things (IoTs) using GA show that finding an exact solution is somewhere difficult as GA uses heuristic approach that yields better solutions. The application of the genetic algorithm is immense and it is widely used in the industries. One such application of GA is the installment of pipe in 3-dimensional space. There are many more related problems like finding the shortest path in a network and the design of a circuit. The complexity of each of the problems is like the traveling salesman problem, which is an np-hard problem that means it's hard to find an exact solution and will require a heuristic algorithm to get a reasonable solution. In this chapter, we have proposed an algorithm that gives significant results for optimizing the 3-D pipe routing in a given space with constraints such as length, cost and number of bends of the pipe. The result shows that in a first iteration the proposed algorithm optimizes the results by approximately 50% in comparison to the existing methods.

V. Maan · A. Malik (✉) · S. Singh
Department of Computer Science & Engineering, Dr. B R Ambedkar National Institute of
Technology Jalandhar, Jalandhar, Punjab, India
e-mail: vivechanam.cs.19@nitj.ac.in; malika@nitj.ac.in

D. Sharma
Department of Computer Science, Kiel University (Christian-Albrechts Universität zu Kiel), Kiel,
Germany
e-mail: des@informatik.uni-kiel.de

Keywords BFS · DFS · GA · NSGA-II · SIMO · SISO · UAV

1 Introduction

Pipe routing problem is a sub problem of path routing problem for internet of things (IoTs) which uses the same approach to create a route from source to destination by adding some additional constraints according to the requirements. The aim in every path routing algorithm is to select the best suited path. Similarly, for pipe routing, same approach of achieving the best suited path is used. Design of pipe routing in 3-d spaces is used to done manually but, it's not practically feasible to explore all the paths manually, which can be designed in given space. Therefore, from decades new methods are introduced regularly to automate the designing of paths in 3-d spaces. While solving this problem, we keep in mind the constraints like cost, length, number of bends etc. that needed to optimize to get the better results at the end. The 3-dimensional pipe instalment problem involves many inlets and outlet pairs ($I1, [O1, O2]$), ($I2, [O3]$), . . . n). It is required to install pipes optimally in all the inlets and outlets by satisfying all the constraints. The pipes can have several branches in them with different sizes of pipe diameters. The 3-d space can be full of obstacles. Traditionally, the path routing algorithm has various applications from traffic optimization to circuit design. Nowadays, by finding a way to optimize the overall path respecting all the constraints helps industries to reduce costs and efforts. There have been many heuristics approaches that have been used to solve the problem. One such approach is the Genetic Algorithm (GA).

A Genetic Algorithm [1] is an evolutionary-based biologically inspired technique. It tries to search global minimal from the overall solution by slow improvement over the previous solution. Genetic Algorithm (GA) has been already proved useful to solve many complex engineering problems. Genetic Algorithm (GA) is mostly used for searching and optimization-related problems, where finding an exact solution is somewhere difficult to find. The application of genetic algorithms is immense and is widely used in the industries. There are various methods which are introduced over the period for path routing problem with the help of GA. There are many more related problems which are somewhere like the 3-d pipe routing problem, like finding the shortest path in-network and design of a circuit on a chip. The complexity of each of the problems is like the travelling salesman problem, which is an np-hard problem that means it's hard to find an exact solution and requires a heuristic algorithm to get a reasonable solution. From the past, routing problems is widely studied because of their various applications in shipping industries, from aero design to large-scale expensive circuit design. The application to this problem is many but due to the nature of the problem and the sheer number of possibilities to design a layout which is both easy to implement and optimal is a very challenging and time-consuming job for even an experienced person. Therefore, it is important to have an automatic algorithmic driven routing layout method.

The 3-d path routing algorithm is not limited to 3-d pipe routing design only but this is a standard problem which is having various applications in different fields. 3-d path routing can be implemented in vast applications such as in computer graphics, designing maps for applications, robot 3-d path planning [2], waterline design, a path for wired networks, networks on chips, 3-d path planning for unmanned aerial vehicles (UAV) [3], a position-based routing algorithm in 3-d sensor networks [4], applying 3-d eco-routing model to reduce the environmental footprint of road transports [5], etc.

Path finding [6] is among one of the most basic yet very important problems. We basically are given the set of points on the map and we have to find the optimal path between those points. It is an important research topic in the area of Computer Science with various applications in fields such as Designing Chips, routing, Strategy Games, GPS and is used in various practical life scenarios. Over time there have been many improvements in a path-finding algorithm which has made it a lot faster and accurate still a very important topic of research for many people just because of the number of applications it has. Artificial Intelligence has shown quite promising results. Many techniques like A* and iterative deepening have been useful in optimizing the algorithm further. Since the coming of the Dijkstra algorithm, a great deal of research has been done in the direction of the Genetic Algorithm for finding the path such that algorithm should be intelligent enough to make its own decision with the change in environment.

Path finding [6] is among one of the most basic yet very important problems. We basically are given the set of points on the map and we have to find the optimal path between those points. It is an important research topic in the area of Computer Science with various applications in fields such as Designing Chips, routing, Strategy Games, GPS and is used in various practical life scenarios. Over time there have been many improvements in a path-finding algorithm which has made it a lot faster and accurate still a very important topic of research for many people just because of the number of applications it has. In particular, Artificial Intelligence has shown quite promising results. Many techniques like A* and iterative deepening have been useful in optimizing the algorithm further. Since the coming of the Dijkstra algorithm, a great deal of research has been done in the direction of the Genetic Algorithm for finding the path such that algorithm should be intelligent enough to make its own decision with the change in environment.

Motivation The motivation to work on this problem is wide use pipe routing designs in industrial uses, pipe routing designs in ships, waterline design in a city, and many more applications. The wide uses of 3-d path routing applications lead to many kinds of solutions to the problem. By reviewing many articles and papers on solutions of 3-D pipe routing algorithms we get to know that the solution for this problem can be enhanced by using genetic algorithms with different kinds of path routing algorithms to diversify the population. We used a genetic algorithm approach for designing 3-D pipe routes to reduce the time for designing the route, generally, the pipe routing is done manually taking in mind all the obstacles, bends, and various constraints depending on the environment and requirement, but

manually this method takes lots of time and overhead. With the help of GA, we can also reduce the cost of pipe routing by minimizing the length of pipes as much as possible, and by reducing the number of bends of pipes. GA can be useful to deal with the path designing of pipes with multiple inlets and outlets which is not easy to handle from other approaches. We choose to work on this problem with the help of GA because it helps to deal with multiple constraints of the problem at the same time, as it works on multiple parameter functions. On the other hand, we used a different kinds of path routing algorithms so, that we can get the best results in different aspects depending on the selected path routing algorithm, which helps us to achieve the optimal solution from the search space.

Problem Definition The Path routing algorithm is closely related to problems like the shortest path algorithm. Therefore, while designing pipe routing in 3-d space most of the time we used the shortest path, low-cost path finding algorithm in a setup. There are multiple cases for this problem such as single-source single destination (SSSD), single-source multiple destinations (SSMD), and finally multiple sources and multiple destinations (MSMD). But in context to our pipe routing problem, MSMD are easily convertible to SSMD. So for simplicity, we will use only SSMD.

Contribution Path routing algorithms are evolving from decades and there are a lot of changes and improvements has been done. Finding the 3-D path routing is quite a challenging task. It not just only has infinite possible ways but also has constraints that we have to satisfy. Our objective is to find the Pareto in front of the given problem. Such that we can choose from all the optimal solutions and no solution is better than the other solution. Based on the facts and description mentioned above, the objectives of this research are as: (1) A method that helps to find the optimal path for multiple input and output points is proposed. (2) it also achieves pareto in front of the given input set. (3) Also applied various path routing algorithms to create solution sets for every generation in GA which helps to get better results.

The rest of the chapter is organized as follows: Sect. 2 discusses the literature review. In Sect. 3 methodology of the proposed work is discussed. Section 4 discusses the performance analysis of the proposed work and the conclusion and future directions are given in Sect. 5.

2 Literature Review

The algorithms to find the shortest path between two points are studied for decades now like Dijkstra's algorithm [7] and A* algorithm. A new improved version of Dijkstra has been created by Hart et al. [8]. The most common path finding maze algorithm was developed by Lee [9]. The algorithm is good for small problems but it is not very effective for large solutions. It does not deal with multiple constraints and does not perform well when there is multiple starting and ending point. Later there

has been a lot of work and improvement done on it by David [10] and Kobayashi [11] but they were not very efficient and could not guarantee an optimal solution. In recent studies, the Genetic Algorithm approach has been taken place in 2018 by Brett et al. [12] but it was not done for multi-branch. In 2016 Wentie [13] introduced the multi-branch approach which proposed the method of a one-point crossover with a maze algorithm for generating the initial population.

Many solutions had been proposed for the path routing problem using various algorithms. Here we discuss some of the genetic algorithm based approaches for pipe routing problems to optimize the design of the pipe routes in an environment using certain constraints.

In 1996 Kim et al. [14] proposed genetic algorithm approach which is used to minimize the length of pipe where interconnections are pre-specified by taking obstacles into account. The results of this paper suggest that GA is superior to either simulated annealing or hill climbing on these kinds of problems.

In 1999 Sandurkar et al. [15] proposed a non-deterministic approach which was based on a genetic algorithm to generate sets of pipe routing with good searching efficiency. Here, STL files are used to represent the obstacles and to show distinctive advantages from them. This approach is useful to optimize the total length and number of bends of connecting pipes while ignoring the obstacles.

In 2005 Shau et al. [16] proposed genetic algorithm approach which is used for the minimization of network cost in a practical environment. The motive is to find optimal pipe design with insufficient time and limited resources. Here gray coding is combined with the improved genetic algorithm with the help of elitist strategy, which means based on fitness degree of offspring. The finding of this study shows that the proposed method is better than the enumeration techniques concerning solution cost and speed.

In 2006 Wang et al. [17] proposed three dimensional multi-pipe route optimization which is done with the help of generalized genetic algorithm. The pipe route has been coded into string using multiple variables and constrain. The algorithm has been claimed to work better than the traditional path finding algorithm for most of the cases. They haven't talk about multi-branch pipe.

In 2009 Ebrahimipo et al. [18] proposed a method in which they have used a weighted cost- based system. The layers are been weighted to show the cost. The efficiency of the genetic algorithm has been compared with the other two, the existing routes and the least cost routes. The genetic algorithm has been proven to be 20% cost-effective. Results showed that least- cost routes and genetic algorithms were producing the same results for most parts.

In 2011 Kusumi et al. [19] proposed a genetic algorithm approach which is used to design pipe path routes in a coal-fired boiler. Here virtual prohibited cells are introduced in search space and a GA-based searching method is used to find multiple paths by avoiding these cells. Virtual prohibited cells are generated with the help of GA to prevent these cells from the random allocation, a sharing method [20, 21] is used in the fitness function to generate these cells. The comparative result shows that the number of patterns with sharing method is 2.5 times more than without sharing method.

In 2015 Wang et al. [22] proposed a method-based genetic algorithm (GA), where the main constraint is distance. The motive is to find a suitable path for pipes that is collision-free. This path is either optimal or near to optimal concerning length, the number of bends, and layout near or away from something. For the distance constraint of the problem, an effective evaluation method is developed for the proposed GA. To evaluate the distance constraint two evaluation functions are added to the evaluation.

In 2016 Niu et al. [13] proposed genetic algorithm is used with the combination of NSGA II (Non-Dominated Sorting algorithm II) and CCNSGA (cooperative Co-evolutionary non-dominated sorting genetic algorithm II) along with rat-maze path finding algorithm to solve the 3-d multi pipe route optimization. This study has used fixed length encoding along with adaptive region strategy.

In 2020 Weihang et al. [23] proposed 3-d pipe routing algorithm which is developed by combining adaptive A* algorithm with genetic algorithm. They showed that it improves the quality of solution. Also with simulation showed that Adaptive A* with genetic algorithm works better than Adaptive A* and A* algorithm.

The above study provides brief information related to the previous study and research carried out in the field of 3d pipe routing using genetic algorithms. The above study shows that the 3d path routing problem is not used for multiple outlets and the option to diversify the population for GA is not considered. This problem of 3d pipe routing can be used as 3d path routing problem which will be best suited in various areas where we need to implement 3d path designs manually. It helps to increase the speed of path design and also helps to achieve optimization. Genetic algorithm uses a heuristic approach that never guarantees to deliver the best result but try to achieve optimization in each iteration by creating new solution sets.

3 Genetic Algorithm

Generally, computer programs can easily be predicted. They start from some point say 1 and after some specific route, they reach point 2. But as development in robotics has shown progress, we get to know that deterministic algorithms are not enough in software, in some situations what is required is the ability to learn and cope with stochastic. When we are looking for the ability of a paradigm of adaptability, we can look for biology as to how it takes place in that. Through years living creatures have been showing their adaptability and flexibility in changing environments. Darwin observed and stated that the procedure of Natural Selection is a method by which evolution happens in the population of specific species. According to this rule evolutionary changes happens to every organism over generations and we can study evolutionary biology to see how evolution happens.

An offspring inherits its feature from parents with the help of genes, and those changes in genes can create new traits in the upcoming generations of an organism. Over generations, a population of organisms obtains so many new traits that it

may become new species. Deoxyribonucleic acid (DNA) plays an important role in evolution with the help of shifting and mixing gene pools.

GA is basically designed to derive the Darwinian theory of evolution with the help of natural selection and biological evolution, therefore it is used to get an optimal or high-performance solution. The main idea behind the genetic algorithm is to initially set a number of solutions that reproduce on their individual fitness in given conditions. The procedure of evolution starts from individuals which are randomly selected from a population. The fitness function is used to calculate the fitness of every individual in each generation. Multiple individuals are selected from generation depending on their fitness level (individuals with the best fitness value are selected) and subjected for crossover, which produces offspring for a new generation. The Fittest members are selected from a pool of population because of good chances of passing their genes successfully to the new generation. The procedure of genetic algorithm consists of six essential steps:

Step 1: We need to set the initial population size for the real problem with the help of that we need to create initial population.

Step 2: Fitness of every individual is calculated depending upon its evolution against the present problem.

Step 3: Check whether the fitness of all individuals is satisfying the criterion of function. If (False)

Step 4: Select parents with higher fitness values for crossover.

Step 5: Crossover and mutation are used to create a new generation, and then replace weak individuals with new solutions. After following this procedure one generation is complete. (Back to step 2)

If (True)

Step 6: Problem is solved.

When dealing with real search problems, we need to take care that search parameters needed to be encoded, and the problem is represented as a function objective which helps to decide the termination of run as objective has been achieved.

A genetic algorithm (GA) consists of the following components: (1) chromosome population, (2) selection depending on the fitness, and (3) crossover to achieve new generation Random mutation of new generation. Now, in the following section, these elements are discussed in more detail.

Chromosome In the GA population chromosomes are represented with the help of a binary string. Every chromosome is present as a point in the search space of all individuals. In the process of GA, we can see a variation that variation in a population of chromosomes that takes place continuously by replacing the old generation with a new generation. Chromosomes are also known as strings, genotypes, individuals, or structures. Representation of chromosomes plays an important role because if it's not flexible then search space gets limited which may make it impossible to find a good solution. As the first step of an algorithm, we need to generate the first set of solutions. Each individual represents the solution to the original problem. These solutions then become the building block of our algorithm.

We will iterate over this solution by slowly progressing and improving the solution little by little at the time. Now the key point here is the solution has to be diverse and should represent the overall solution. The more the diverse initial population will be the better a chance we will reach the global optimal solution.

Fitness Assessment The fitness level of each individual is assessed with a value that is calculated with the help of the fitness function. The fitness of each individual is represented as the ability of that individual to solve the problem. Most commonly genetic algorithms are used for optimizing the given problem, whose motive is to find a set of parameters that either minimize or maximize a complex multi-parameter method. Fitness function takes population as input and tells the fitness score of each of the chromosomes. Now the fitness score is dependent on what we need in the solution. Based on all the constraints it is to map the fitness score to each chromosome which is the identifier of how good the chromosome solution is. Based on the fitness score the probability of getting a chance to reproduce also increases.

Selection The selection operator is used to select individuals from a current population for crossover, the individuals with higher fitness values are having higher chances for selection. The selection operator is designed such that some amount of fewer fit individuals can get a chance to get selected. This property helps to maintain diversity in the population, therefore preventing premature convergence on less fit solutions. The idea between doing the selection is choosing two-parent and passes their genes to the next generation. Now based on all the fitness scores we get there are several ways of selecting the parent. In general, we give more chances to the fitter chromosome this thing is known as survival of the fittest. We assign a probability to each chromosome based on the fitness score and then based on probability we randomly choose parents and do crossover to produce their offspring. This method works under the assumption that the best solution will be around the fitter solution. Here, we discuss two methods of selection which are roulette wheel selection and tournament selection.

- **Tournament selection:**

Tournament selection is having two steps: In the first step, we need to select a group of N chromosomes ($N \geq 2$), now in the second step we need to select an individual from this group we need to select the individual with greatest fitness value for crossover and mutation, while all other individuals will be there in the gene pool for selection in next loop. The selection procedure can be handled with the help of tournament size. So, tournament size is inversely proportional to the chance of getting selected, e.g. with a smaller tournament size the chance of selection for individuals with smaller fitness value increases. Generally, once an individual is selected then it will be discarded from the pool for further selection to maintain the diversity in the new generation.

- **Roulette wheel selection:**

This is a selection method that is used in GA to select a solution that is potentially useful for crossover. Individuals with higher fitness values are having higher chances for selections, but there may still be a chance to not get selected. The advantage of this selection method is that there may be a chance for weaker individuals to get selected for crossover. The individual who is weaker in the present generation may get some useful traits by recombining with some strong individuals, and with the help of these selection criteria, those traits can be passed to the upcoming generations.

This selection method can be imagined like each individual is represented as a pocket on the wheel. The size of this pocket depends on the fitness of an individual and the probability of selection. The probability of selection of N individuals from the population pool is similar to that of playing N number of games on the roulette wheel as each individual is selected independently.

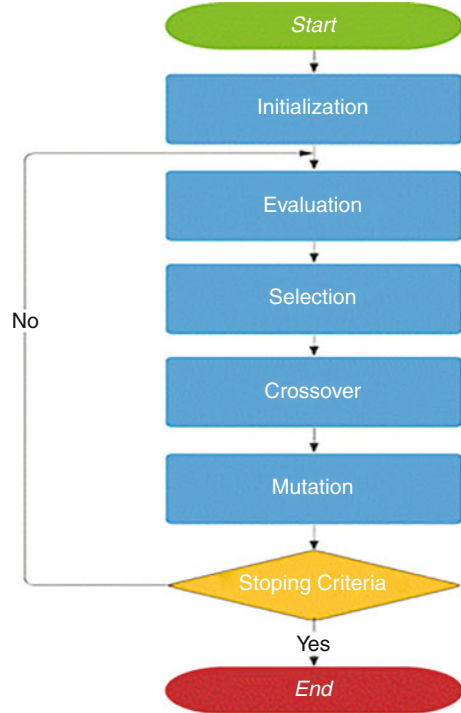
Crossover

It is the most vital part of the algorithm. The crossover point is chosen at random from the selected parents and the genes are exchanged from them so to create their off-springs which contain genes of both the parents. Now their children's are added to next generation. The crossover operator is the key to the power of GA. There are various types of crossover methods such as uniform crossover or n-point crossover. This operator works based on biological recombination. With the help of crossover, the offspring can have the best traits from both parents. However, a crossover is beneficial to create the best from the two individuals, but it may also happen that it breaks up the good individual, but with the help of selection, we may overcome this problem in GA. Crossover shows negative impacts when only the best individuals get selected all the time, it will lead to breaking the good individual in the procedure.

Mutation

The mutation is a slight variation in one or two genes of the children population to maintain the diversity in the population and not letting solution converging prematurely and ending u with local minima/maxima instead of global minima/maxima. The mutation is done with a lower probability. This process is repeated until the desired solution is achieved. This method is used to select a point randomly in the bit-string and then change the value of that point either from 1 to 0 or from 0 to 1. This procedure helps to maintain the population more diverse. With small possibility, mutation can take place at each bit position but in this way, we may affect mutation intensively. Here, we also use the Pareto front to find the solution. The Pareto front is a set of non-dominated solutions where no solution is better than the other, i.e. no one parameter can be improved without at least sacrificing one or more parameters. The complete process of the genetic algorithm is given in Fig. 1.

Fig. 1 Flow diagram for genetic algorithm



4 Methodology

The 3-dimensional pipe installment problem involves many inlets and outlet pairs. We have to install pipes optimally in all the inlets and outlets by satisfying all the constraints. Pipes can have several branches with different sizes of pipe diameters.

4.1 Representation of 3-Dimensional Space

Now the representation of space in three-dimensional with all the detail is a very complex and time-consuming task. In addition, too much detail of the 3-d mapping is might be insignificant for most of the practical use. So now to get a reasonable amount of detailed mapping of the 3- dimensional mapping, divide the 3-dimensional space into a small grid-like structure. The advantage of this kind of structure is that we can mathematically formulate it and the representation of space gets hugely simplified. The disadvantage of the grid system is we lose some information but with this representation, the exactness of the model can be easily controlled by increasing or decreasing the size of cubes.

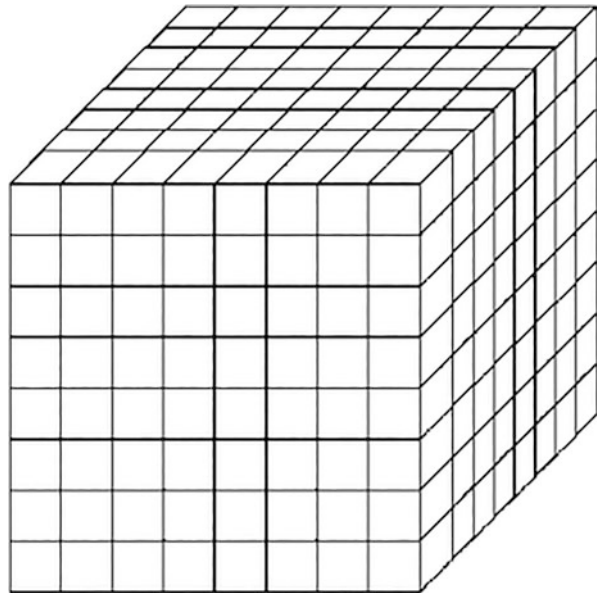
4.2 *Mathematically Modelling of 3-d Space*

To represent the 3-d space of variable size 3-d space we have to represent it with a fixed size of a cube in approximately the best fit for our 3-d space. So the (l_m_n) size of the cube is being approximated by the (L_M_N) size of 3-d space such that we can get integer no of cubes in all x, y, z directions. Each cell of the 3-dimensional space is represented by the coordinate system such as (1, 1, 1) and so on. We will consider x equals y equals z ($x == y == z$), otherwise, we cannot make straight pipes. So suppose we change the direction in a pipe that is diagonal. In that situation, the angle of the pipe is difficult to determine. Therefore, we will only consider that pipe can bend either 90 or 45°. The size of the cube depends on the level of accuracy we want in our solution. The more dense cubes will produce a better solution but on the other hand, it will increase the complexity of the problem. So we will try to choose the size of the box depending upon the kind of obstacle we have as shown in Fig. 2.

4.3 *Representation of Obstacles*

Showing the obstacle in a grid-like structure is easy. Each cell of the cube can be represented as 0 and 1 where 0 means free space and 1 means block(obstacle) in our 3-dimensional space. Any small or partial part of the obstacle can be considered

Fig. 2 Divided 3-d space into grid



as the block part and we can fill 1 in that block of the grid. In this way, we can approximately outline all the obstacles of the 3-d space where the pipe cannot go.

The other advantage of representing the obstacle as 0 and 1 is we can easily control the area of the path routing algorithm. So suppose we have an area where we don't have obstacles but we still don't want our routing algorithm to set the pipe in those areas. We can easily eliminate that problem by putting 1 in that block as shown in Figs. 3 and 4. This method is also useful in blocking the path of other paths which has been already created.

Fig. 3 3-d space with circle as obstacle

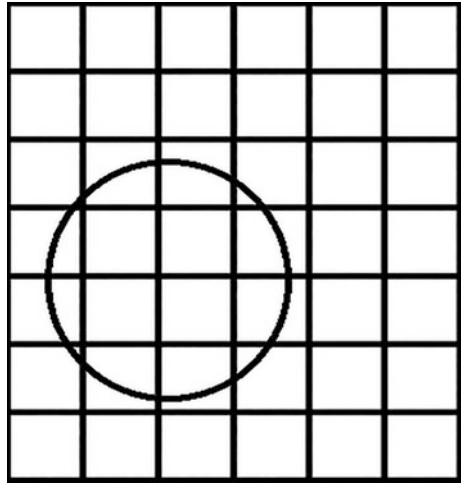
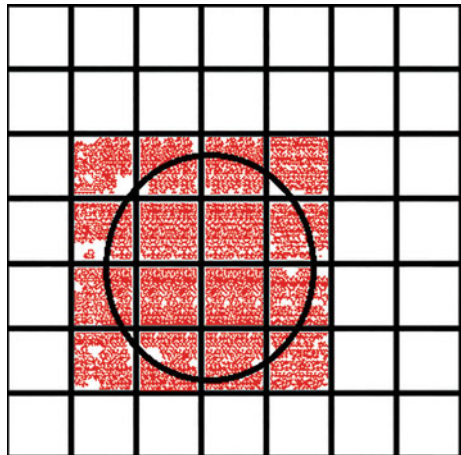


Fig. 4 3-d space with red region as blocked grid



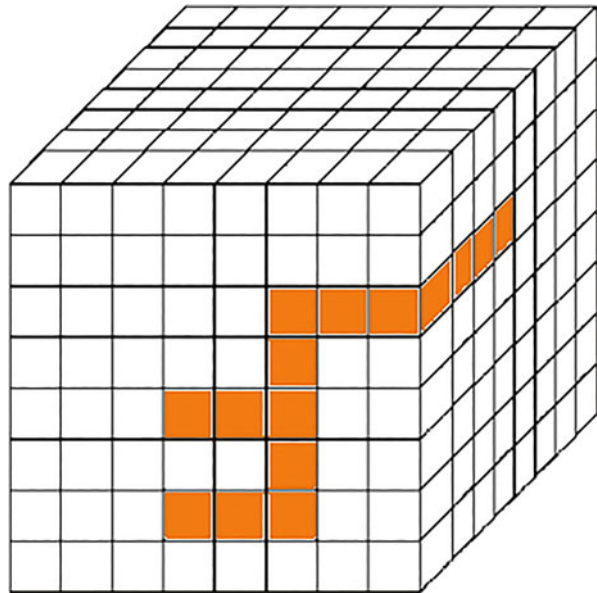
4.4 Representation of Pipe

Now in 3-dimensional space to represent the pipe, it needs to be in a continuous grid or can make an L and T shape in a 3-d space. Where L represents bend in the pipe and T represents the junction between the two inlets and outlet pipe, i.e. other than, 90° there are no turns.

4.5 Problem Formulation

Pipe routing problem is a multivariate optimization problem in which there are multiple parameters we want to satisfy or optimize to the best of our efforts as shown in Fig. 5. In general, these constrain can be anything that can be mathematically quantifiable. The most common constrain for this problem are: We cannot pass through the obstacle (1) the number of bends overall pipe can take should be globally minimized, (2) the total length of pipe should be minimized, (3) maximum optimization should be there between the branched-out pipes, and (4) two different pairs of inlet and outlet pipes should not intersect each other.

Fig. 5 Branched pipe in 3-d space



4.6 Implementation

There are two variants of the problem which we want to solve

- **Single Inlet Single Outlet:** In this, we will have a single source of input and a single source of output.
- **Single Inlet Multiple Outlet:** In this, we will have a single source of inlet and multiple sources of outlet. In other words, we can have branches in the pipe.

4.6.1 Single Inlet Multiple Outlet

This case is a little bit complicated than the Single inlet single outlet (SISO). So we are given with the set on Input (I1, I2, . . . , In) and set of Output ([O11, O12 . . . , O1n], [O21, O22, . . . , O2n], . . . , [On1, On2, . . . , Onn]). We need to connect each input to its respective output such that we obey all the constraints. In this case for every inlet, there is an array of outlets. This makes this case more complicated. So to overcome this what we do is we map each inlet with every outlet and try to find a path between them. We also introduce one more fitness constrain i.e. maximum overlap. This function will essentially check how much overlap is there between a single inlet and multiple outlet pair.

• Creating Population

Firstly, we need to generate the initial population. For that, we will again use any path-finding algorithm such as A* or maze algorithm. For diversity again we will introduce all the algorithms. So, for branching pipes, we make our path finding algorithms further refined to handle that case. Now take an example of pair I1 as one inlet and O1, O2, O3 as the three-outlet point. In cases like this first, we will find a path between I1 and O1. In this case, notice that I1 and O1 are the starting and terminating points for the path-finding algorithm. Suppose the path which we got was I1, p1, p2, p3, p4, p5, O1. Now for finding the branch to O2 we can consider all the previous points in the path as a starting point. This might look strange, but it helps us to get maximize overlapping. So now the path which we get is I1, p1, p2, p6, p7, O2. Now similarly for O3 we can consider all point I1, p1, p2, p3, p4, p5, O1, p6, p7, O2 as the starting point and O3 as terminating point. In this way, we can generate all the initial solutions using a combination of different algorithms and the technique discussed above.

• Evaluation

We will find the fitness matrix of each chromosome in our population. For that, we mathematically quantify a good solution. For this problem, we will consider length and number of bends, and maximum overlapping as the optimization criteria. We try to decrease all. The fitness function gives a fitness score to each individual chromosome, which shows us how to fit the individual solution. The selection of an individual depends upon the fitness score of the individual. In our fitness function,

we simply assign a cost with the bend and length and subtract the length of the overlapping pipe. Suppose we have taken the cost of length x unit and bend be y unit. So, the total cost of a path will be equal to $(\text{length} \times x + \text{Number of bends} \times y - \text{overlapping length})$. The overall cost of the chromosome will be an addition of all the inlet and outlet paths. In this case, we need maximum overlapping between inlet point $I1$ and outlet point $O1, O2 \dots$ and so on. So the path from $I1$ to $O1$ and path from $I1$ to $O2$ should have maximum overlapping as it reduces the amount of length of pipe is required. Note that we don't want to overlap between different pairs of pipes.

• **Selection**

Based on all the constraints and getting fitness of the chromosome we will sort further. Now for sorting the chromosomes based on their fitness matrix we use NSGA-ii (Non-dominated sorting) algorithm. After sorting each chromosome the next step is a selection of the chromosome. The idea behind this step is to pass their genes to the next generation. So, the fittest individual gets the higher priority of getting chosen and reproduces. So, we randomly choose two chromosomes with the probability assign to them for creating new off-springs.

• **Crossover**

Two parents are selected randomly based on their probability and have crossover and mutation as shown in Fig. 6 and 7. Now for crossover we make the group of inlet outlet pair for e.g. if there are inlet $I1$ and outlet $O1$ and from Parent one we get path $P1[(I1, \dots, O1)]$ and for parent two we get $P2[(I1, \dots, O1)]$. From Parent one we randomly select one point p' randomly suppose parent one $P1$ denoted as $P1p'$ and similarly one point p'' randomly suppose parent two $P2$ denoted as $P2p''$. Now from $P1p'$ to $P2p''$ find new path. Suppose the new path is $(P1p', \dots, P2p'')$. Then the children solution can be $(P1I1, \dots, P1p', \dots, P2p'', \dots, P2O1)$ and $(P2I1, \dots, P1p', \dots, P2p'', \dots, P1O1)$ as shown in Fig. 8.

• **Mutation**

The next step is mutation and it is done so as to give diversity to our solution set and not letting premature convergence of the solution. As premature, convergence may lead to local minima solution and not global minima. Mutation helps the solution to go to unexpected solutions and search over there. Mutation should not be done with a higher probability as it results in slow convergence of the solution. The way traditionally mutation is done by changing one bit randomly cannot be done here

Fig. 6 Parent 1 chromosome (SIMO)

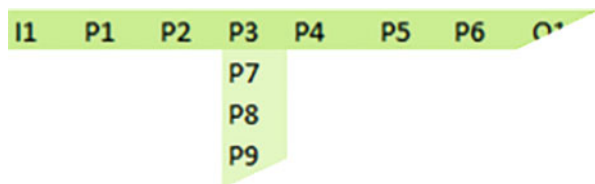


Fig. 7 Parent 2 chromosome (SIMO)

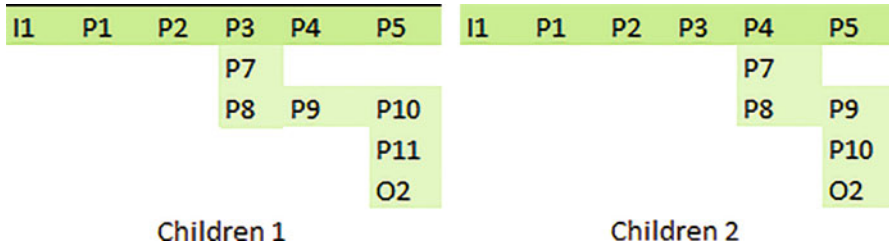
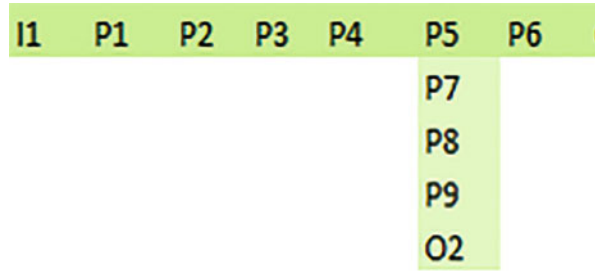


Fig. 8 Children one and two chromosome made by crossover



Fig. 9 Children one and two chromosomes made by crossover

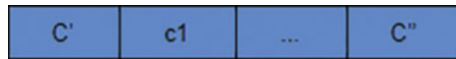


Fig. 10 Children one and two chromosomes made by crossover



Fig. 11 Children one and two chromosomes made by crossover

as we cannot guarantee the correctness of the path. A mutation like this will lead to desecrate pipe formation. To resolve this problem we can randomly choose 2 points from the children to suppose (c1, c2) and generate a path from our random path generating algorithm now this path can be replaced from the children with the newly generated path. Now in Fig. 9, we can see that C' and C'' a path from (C1, Cn) are chosen at random.

The point C' and C'' should not be much closer to each other as they can potentially change the complete property of children's. Figure 10, shows the random path generated from the points C' and C''. This path can now be used as the mutation in the chromosome.

Figure 11 shows the mutated children.

Now from the number of chromosomes (Both parents and children), we find the fitness score and sort using NSGA-ii. The fittest chromosome is selected and the rest of the solution is thrown away. This process is iterated over a number of generations till the solution is converged.

4.6.2 Single Inlet Multiple Outlet

This case is a little bit complicated than the Single inlet single outlet (SISO). So we are given with the set on Input (I_1, I_2, \dots, I_n) and set of Output ($[O_{11}, O_{12}, \dots, O_{1n}], [O_{21}, O_{22}, \dots, O_{2n}], \dots, [O_{n1}, O_{n2}, \dots, O_{nn}]$). We need to connect each input to its respective output such that we obey all the constraints. In this case for every inlet, there is an array of outlets. This makes this case more complicated. So to overcome this what we do is we map each inlet with every outlet and try to find a path between them. We also introduce one more fitness constrain i.e. maximum overlap. This function will essentially check how much overlap is there between a single inlet and multiple outlet pair.

• Creating Population

Firstly, we need to generate the initial population. For that, we will again use any path-finding algorithm such as A* or maze algorithm. For diversity again we will introduce all the algorithms. So, for branching pipes, we make our path finding algorithms further refined to handle that case. Now take an example of pair I_1 as one inlet and O_1, O_2, O_3 as the three-outlet point. In cases like this first, we will find a path between I_1 and O_1 . In this case, notice that I_1 and O_1 are the starting and terminating points for the path-finding algorithm. Suppose the path which we got was $I_1, p_1, p_2, p_3, p_4, p_5, O_1$. Now for finding the branch to O_2 we can consider all the previous points in the path as a starting point. This might look strange, but it helps us to get maximize overlapping. So now the path which we get is $I_1, p_1, p_2, p_6, p_7, O_2$. Now similarly for O_3 we can consider all point $I_1, p_1, p_2, p_3, p_4, p_5, O_1, p_6, p_7, O_2$ as the starting point and O_3 as terminating point. In this way, we can generate all the initial solutions using a combination of different algorithms and the technique discussed above.

• Evaluation

We will find the fitness matrix of each chromosome in our population. For that, we mathematically quantify a good solution. For this problem, we will consider length and number of bends, and maximum overlapping as the optimization criteria. We try to decrease all. The fitness function gives a fitness score to each individual chromosome, which shows us how to fit the individual solution. The selection of an individual depends upon the fitness score of an individual.

In our fitness function, we simply assign a cost with the bend and length and subtract the length of the overlapping pipe. Suppose we have taken the cost of length x unit and bend be y unit. So, the total cost of a path will be equal to $(\text{length} \times x + \text{Number of bends} \times y - \text{overlapping length})$. The overall cost of the

chromosome will be an addition of all the inlet and outlet paths. In this case, we need maximum overlapping between inlet point I1 and outlet point O1, O2 ... and so on. So the path from I1 to O1 and path from I1 to O2 should have maximum overlapping as it reduces the amount of length of pipe is required. Note that we don't want to overlap between different pairs of pipes.

• **Selection**

Based on all the constraints and getting fitness of the chromosome we will sort further. Now for sorting the chromosomes based on their fitness matrix we use NSGA-ii (Non-dominated sorting) algorithm. After sorting each chromosome the next step is a selection of the chromosome. The idea behind this step is to pass their genes to the next generation. So, the fittest individual gets the higher priority of getting chosen and reproduces. So, we randomly choose two chromosomes with the probability assign to them for creating new off-springs.

• **Crossover**

Two parents are selected randomly based on their probability and have crossover and mutation. Now for crossover we make the group of inlet outlet pair for e.g. if there are inlet I1 and outlet O1 and from Parent one we get path P1[(I1, ...,O1)] and for parent two we get P2[(I1, ...,O1)]. From Parent one we randomly select one point p' randomly suppose parent one P1 denoted as P1p' and similarly one point p'' randomly suppose parent two P2 denoted as P2p''. Now from P1p' to P2p'' find new path. Suppose the new path is (P1p', ..., P2p''). Then the children solution can be (P1I1, ..., P1p', ... P2p'', ... P2O1) and (P2I1, ..., P1p', ... P2p'', ... P1O1) as shown in Figs. 12, 13 and 14.

Fig. 12 Parent one chromosome (SIMO)

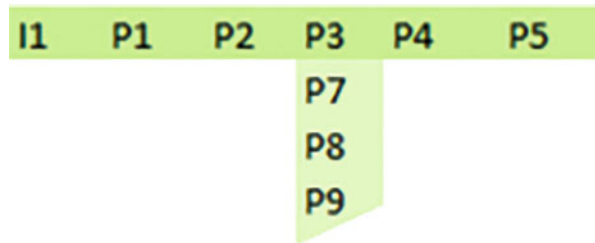
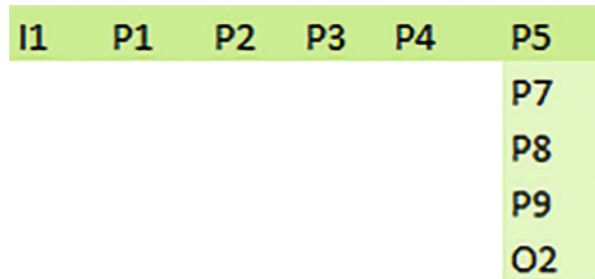


Fig. 13 Parent two chromosome (SIMO)



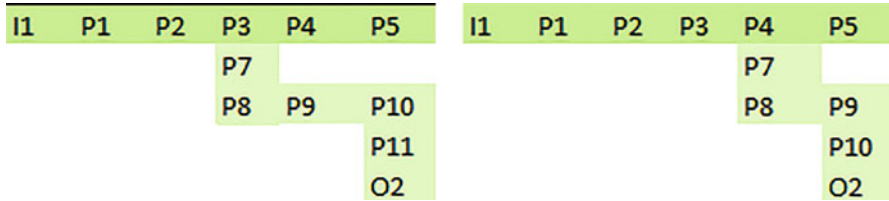


Fig. 14 Children one and two chromosome made by crossover



Fig. 15 Children one and two chromosomes made by crossover

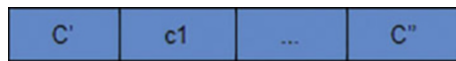


Fig. 16 Children one and two chromosomes made by crossover



Fig. 17 Children one and two chromosomes made by crossover

• **Mutation**

The next step is mutation and it is done so as to give diversity to our solution set and not letting prematurely convergence of the solution. As premature, convergence may lead to local minima solution and not global minima. Mutation helps the solution to go to unexpected solutions and search over there. Mutation should not be done with a higher probability as it results in slow convergence of the solution. The way traditionally mutation is done by changing one bit randomly cannot be done here as we cannot guarantee the correctness of the path. A mutation like this will lead to desecrate pipe formation. To resolve this problem we can randomly choose 2 points from the children to suppose (c1, c2) and generate a path from our random path generating algorithm now this path can be replaced from the children with the newly generated path. Now in Fig. 15 we can see that C' and C'' a path from (C1, Cn) are chosen at random.

The point C' and C'' should not be much closer to each other as they can potentially change the complete property of children's. Figure 16, shows the random path generated from the points C' and C''. This path can now be used as the mutation in the chromosome.

Figure 17 shows the mutated children.

Now from the number of chromosomes (Both parents and children), we find the fitness score and sort using NSGA-ii. The fittest chromosome is selected and the rest of the solution is thrown away. This process is iterated over a number of generations till the solution is converged.

5 Results and Discussion

Different path routing algorithms are used in creating a solution set for GA so, that we can have different kinds of benefits in the solution set depending upon the path routing algorithm for that particular iteration. GA uses a heuristic approach to achieve optimal solutions. The two algorithms can always work better than each other based on the condition it is run on or constraints applied on it. It is especially true in our case. But still, to have a fair comparison we implemented Wentie Niu et al. [13] algorithm to the best of our knowledge and then the same input was feed to the algorithms.

5.1 Base Environment

We created a $100 * 100 * 100$ dimension 3d grid as our working area of the internet of things. We randomly put some obstacles in it. The obstacle in the grid is represented as 1 and free space as zero. The working area remains constant throughout the experiment. The inlet and outlet points also remain same for both experiment.

5.2 Performance Study

For a similar environment, we change two parameters first is the number of generations, and the second is the number of population.

5.3 Number of Generations

It has been noticed that a number of generation's increases in both the solution coincide. This is logical to happen as we run the algorithm sufficiently enough, no matter which algorithm we use it should reach the global optimal. But in our case the solution we designed converged to the solution faster than the existing methods [13] solution (Table 1 and Fig. 18).

Table 1 Comparison between Niu et al. [13] method vs. proposed method

Generation	Niu et al. [13] method	Proposed method
0–5	327–184	189–153
5–10	184–173	153–149
10–15	179–161	149–143
15–20	161	143–141

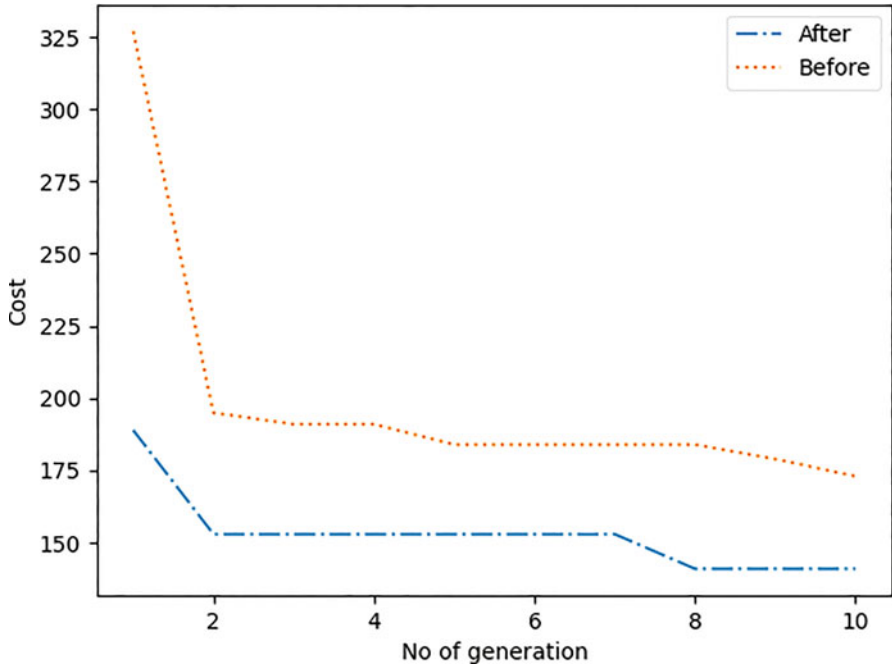


Fig. 18 Number of generations

Table 2 For 2-d room,
Population size: 10, Room
size: 20*20, Best answer: 41

Generation	Best cost in generation
0–9	65–57
10–19	53
20–29	49
30–39	47
40–49	43
50–59	43

5.4 Number of Populations

The trend seen with the increase of population was dicey. There was nothing concretely we could have said which one is better. In a few situations Niu algorithm [13] had a better result and in a few our implementation. But we still can choose our implementation due to the kind of control is provided to the proposed method. We had a more generalized solution. We also noticed that with the increase in population. The converging rate of both the solution increased. Again it’s a logical conclusion as the number of solutions increases per iteration greater is the probability of finding the global optimal solution. Hence, see better converging rate (Tables 2 and 3).

Table 3 Generation fix: 10, Population vary: 10

Generation	Population size	Starting cost	Best cost in generation
0–9	10	75	53
0–9	20	67	47
0–9	30	77	47
0–9	40	65	43

6 Conclusion and Future Scope

In this work, we provided a detailed overview of 3-D pipe routing techniques based on a genetic algorithm. We see how with the help of GA we can make the pipe routing modal time-efficient, cost-efficient with limited resources. With the help of GA, we achieved near to optimal solutions by considering different parameters. Here, we discuss the basic functionality of the genetic algorithm and see its workflow. With the help of GA, we can fully automate the pipe routing process which is manually very time consuming, it's not possible to explore all routes manually present in a given domain but with GA we can search all solutions present for pipe routes and pick the best suited for the given situation. Modification in GA can be done by merging different algorithms with it which can help to yield a better solution.

We have used a mix of a lot of path-finding algorithms all together each brings its unique nature to the solution. But overall the only thing which makes a difference is how diverse the solution is. Though it was noticed sometimes a very unique solution which at first doesn't look like a good solution ends up becoming the best solution in many interesting ways. Though we have tried with only two parameters it is interesting to see how people tackle problems like serviceability and a different size of pipe and so on.

Future work can be done to diversify the population using different kinds of path-finding algorithms. By doing so we take benefit of all the path finding algorithms which help to maintain diversity in the population even further. This diversified population increases the probability to get the optimal solution.

References

1. Yang, X.-S. (2021). *Genetic algorithms, nature-inspired optimization algorithms* (2nd ed., pp. 91–100). Academic. Chapter 6.
2. Liang Yang, Juntong Qi, Dalei Song, Jizhong Xiao, Jianda Han, & Yong Xia. (2016). Survey of robot 3D path planning algorithms. *Journal of Control Science and Engineering*, 2016. <https://doi.org/10.1155/2016/7426913>
3. Omar, R., & Gu, D.-W. (2010). 3D Path Planning for Unmanned Aerial Vehicles using Visibility Line based Method, *I*, 80–85.
4. Al Tahan, A. M., & Watfa, M. K. (2012). A position-based routing algorithm in 3D sensor networks. *Wireless Communications and Mobile Computing*, 12, 33–52.

5. Busho, S. W., & Alemayehu, D. (2020). Applying 3D-eco routing model to reduce environmental footprint of road transport in Addis Ababa City. *Environmental Systems Research*, 9, 17.
6. Path routing algorithm <https://neo4j.com/developer/graph-data-science/path-finding-graph-algorithms/>
7. Dijkstra, E. W., et al. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269–271.
8. Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107.
9. Chin Yang Lee. (1961). An algorithm for path connections and its applications. *IRE Transactions on Electronic Computers*, 3, 346–365.
10. David, W. (1969). Hightower. A solution to line-routing problems on the continuous plane. In *Proceedings of the 6th annual design automation conference*, pp. 1–24.
11. Kobayashi, Y., Yoshida, K., & Yoshinaga, T. (1993). A knowledge-based approach to verification and improvement of industrial plant layout design. In *Expert systems in engineering applications* (pp. 179–189). Springer.
12. Wanamaker, B., Cascino, T., McLaughlin, V., Oral, H., Latchamsetty, R., & Siontis, K. C. (2018). Atrial arrhythmias in pulmonary hypertension: Pathogenesis, prognosis and management. *Arrhythmia & Electrophysiology Review*, 7(1), 43.
13. Niu, W., Sui, H., Niu, Y., Cai, K., & Gao, W. (2016). Ship pipe routing design using NSGA-II and coevolutionary algorithm. *Mathematical Problems in Engineering*, 2016, 1–21.
14. Kim, D. G., Corne, D., & Ross, P. (1996). Industrial plant pipe-route optimisation with genetic algorithms. In *International conference on parallel problem solving from nature* (pp. 1012–1021). Springer.
15. Sandurkar, S., & Chen, W. (1999). Gaprus genetic algorithms based pipe routing using tessellated objects. *Computers in Industry*, 38(3), 209–223.
16. Shau, H.-M., Lin, B.-L., & Huang, W.-C. (2005). Genetic algorithms for design of pipe network systems. *Journal of Marine Science and Technology*, 13(2), 116–124.
17. Wang, H., Zhao, C., Yan, W., & Feng, X. (2006). Three dimensional multi-pipe route optimization based on genetic algorithms. In *International conference on programming languages for manufacturing* (pp. 177–183). Springer.
18. Ebrahimipo, A. R., Alimohamad, A., Alesheikh, A. A., & Aghighi, H. (2009). Routing of water pipeline using GIS and genetic algorithm. *Journal of Applied Sciences*, 9(23), 4137–4145.
19. Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT Press.
20. Deb, K., & Goldberg, D. E. (1989). An investigation of niche and species formation in genetic function optimization. In *Proceedings of the third international conference on Genetic algorithms*, pp. 42–50.
21. Goldberg, D. E., Richardson, J., et al. (1987). Genetic algorithms with sharing for multimodal function optimization. In *Genetic algorithms and their applications: Proceedings of the second international conference on Genetic algorithms* (pp. 41–49). Lawrence Erlbaum.
22. Wang, C., Sun, X., & Yuan, T. (2015). A method based genetic algorithm for pipe routing design. In *2015 international conference on advanced engineering materials and technology* (pp. 826–830). Atlantis Press.
23. Lv, W., Qin, N., Zhao, X., Yuan, P., & Huang, J. (2020). Pipe routing of reactor based on adaptive a* algorithm combined with genetic algorithm. In *2020 Chinese Automation Congress (CAC)* (pp. 5567–5572). IEEE.

Part III
Further Reading

Intelligent Fog-IoT Networks with 6G Endorsement: Foundations, Applications, Trends and Challenges



Syed Anas Ansar, Jitendra Kumar Samriya, Mohit Kumar,
Sukhpal Singh Gill , and Raees Ahmad Khan

Abstract The prolonged 5G network deployment includes the Internet of Things (IoT) as a technological advancement toward the expansion of wireless communication. The Internet of Everything (IoE), a superset of IoT, acts as the proliferation that accelerated the outburst of data and sparked new disciplines. Nonetheless, the foundational and crucial elements of an IoE depend heavily upon the computing intelligence that could be implemented in the 6G wireless communication system. This study aims to demonstrate the 6G-enabled fog architecture as a rigorous integrated IoT solution designed to accommodate seamless network operations and management. Fog computing (FC) is a game-changing technology that has the potential to deliver data storage and computation capabilities to forthcoming 6G networks. In the 6G generation, fog computing will be essential to support gigantic IoT applications. In recent years, the amount of IoT-linked nodes and gadgets in our everyday lives has increased rapidly. Fog computing has evolved into a well-established framework for addressing a wide range of critical Quality of Service (QoS) criteria, including latency, response time, bandwidth constraints, flexibility, security, and privacy. In this manuscript, the research explored 6G networks with IoT and fog computing technology in depth. This article outlines fog-enabled intelligent IoT applications while emphasizing the IoT networking context. The

S. A. Ansar
Babu Banarasi Das University, Lucknow, India

J. K. Samriya
Graphic Era Deemed to be University, Dehradun, India

M. Kumar
NIT Jalandhar, Jalandhar, Punjab, India
e-mail: kumarmohit@nitj.ac.in

S. S. Gill
School of Electronic Engineering and Computer Science, Queen Mary University of London,
London, UK
e-mail: s.s.gill@qmul.ac.uk

R. A. Khan (✉)
Babasaheb Bhimrao Ambedkar University, Lucknow, India

main objective of this study is to embrace varying technologies to elucidate notions, including modern IoT applications that exploit fog in Beyond fifth-generation (B5G) and 6G networks. Thus, it addresses specific issues and challenges that IoT may stumble into and implies potential fog solutions.

Keywords Fog computing (FC) · Internet of Things (IoT) · 6G network · Quality of Service (QoS) · Wireless communication

1 Introduction

Pervasive or ubiquitous computing, which refers to computing that takes place anytime and anywhere, dates back to the 1990s when gadgets and wireless networks were not quite as advanced as they are now. Hence, it can be inferred from the above statements that technology advances at a breakneck pace. According to a recent edition of *The Economist*, “by 2035, the entire planet will have a trillion networked computers embedded into everything from garments to food packaging and construction industry” [1]. The IoT has swiftly become a strategic change factor for all businesses because it merges the physical and digital realms of the modern world. It offers a slew of advantages, which include a new generation of smart, linked gadgets. New value products are produced across various industries in the networked society by linking objects and imparting “acumen” to them. IoT can be described as a vast network of physical objects, such as security systems or household appliances, connected to the internet for convenient utilization. Digital components like sensors, microprocessors, software, connectivity, and data storage are extensively used in these products, besides mechanical and electrical components. As the world’s physical and digital edifices decussate, digital technologies are built into an array of consumer and industrial items; for example, motion sensors and video cameras are a part of internet-connected smart doorbells that warn a homeowner when someone approaches the door. The homeowner may monitor and converse with the guest via a smartphone application, while a video of their conversation is preserved for further protection. The demand for Internet of Things (IoT)-based linked applications and devices like smartphones, Google Glass, gadgets, etc., has accelerated immensely in recent years [2]. A tremendous amount of data is being produced rapidly due to the expansion of millions of IoT-linked devices. As a result, cloud storage becomes increasingly constrained as the amount of data produced, stored, and managed soars [3]. The cloud computing servers connect these devices in a hectic way, causing a slew of problems in optimizing important Quality of Service (QoS) characteristics, including processing, privacy, bandwidth, security, latency, response time, and storage [4, 5]. Since the cloud acts as a centralized mainframe to compute and store data and is generally located remotely from the IoT endpoints, the cloud server may take some time to respond to the data. Therefore, emerging fog computing technology relieves the strain of cloud computing services. Fog computing is a decentralized computer framework that distributes storage,

intelligence control, and processing among data devices in close proximity. This framework now makes cloud computing services available at the network's edge. Axiomatically, the network distance is reduced, and the quantity of data that has to be transferred to the cloud for processing, analysis, and storage is also reduced [6–12]. Wireless communication technologies have grown at an astounding rate, catapulting the way machines and humans interact. The enormous expansion of connected devices, together with the ever-increasing necessity for large bandwidth, have been the primary driving catalyst for such evolving advancements during the last decade. Though the implementation of 5G wireless communications is in its nascent stages with enmeshed characteristics that need improvement, it has become critical to consistently determine the future communication requirements and begin theoretical and practical projects on futuristic wireless system development. In general, any advancement in successive generations occurs over a 10-year period. 6G is expected to become widely available by 2030 [13]. The 6G market is expected to enable prominent advancements in imaging, location awareness, and presence technologies. With the collaboration with Artificial Intelligence (AI), the 6G computational infrastructure shall decide the ideal location for computing, including decisions concerning processing, data storage, and sharing. A significant aim of the 6G network is to facilitate communications with one-microsecond latency, which is a thousand times greater than 1-millisecond throughput (or 1/1000th of the latency). 6G is expected to deliver 1000 times the number of simultaneous wireless connections as 5G technology. Hence, the latency and capacity of 6G applications will be improved. It will also enable new and innovative wireless networking, cognitive, sensing, and imaging applications. 6G access points will be able to serve numerous clients at the same time using orthogonal frequency-division multiple accesses. The advent of smartphones increased the use of 3G services and encouraged 4G deployment requirements in the IoT business model; it is expected that some IoT businesses would support the 5G outbreak somewhere during the 5G period, hence increasing demand for upcoming 6G networks. Large IoT networks equipped with 6G networking and cognitive learning algorithms can swiftly conduct complicated calculations, transforming the user experience to real-time responsiveness [14]. Massive IoT networks will transform the healthcare, transportation, agricultural, and business sectors. Also, Smart grids with interconnected energy, water, and gas lines will make cities intelligent [15, 16]. Besides this, autonomous vehicles and wearable devices will become commonplace, making lives smoother and more convenient [17–19]. The complexity of a network grows in lockstep with its capacity. Heterogeneity, integration, interoperability, network capacity, network congestion, scalability, QoS provisioning, and battery lifespan are a few of the challenges that will be faced by 6G-empowered IoT [20–22]. IoT will be contingent on intelligent learning approaches and the extensive deployment of edge and fog computing devices in close proximity to end devices in order to overcome the challenges above [23, 24]. By performing calculations closer to end devices, edge and fog computing devices will alleviate the pressure on cloud servers, reducing computing latency [12, 25]. To boost the network efficiency even further, fog devices will smartly commingle spare resources from all accessible devices [26, 27]. The



Fig. 1 Expansion of IoT in different domains

processing resources of accessible fog devices, edge devices, and other devices will be essential in living up to the expectations of highly demanding future applications (Fig. 1).

1.1 Article Organization

The remaining article is structured as shown in the following sections: Sect. 2 presents a literature review by different authors in this field. An overview of the three main subjects of discussion that are in the spotlight throughout this paper is given in Sect. 3. In Sect. 4, several applications of 6G-enabled Fog IoT networks are presented. Furthermore, Sect. 5 presents possible fog solutions for particular IoT challenges. Last but not least, the paper binds up in the next section.

2 Related Studies: Current Status

Ananya Chakraborty et al. outlined the progression of computing paradigms from the client-server model to edge computing, along with its goals and constraints. An up-to-date analysis of cloud computing and the cloud of things (CoT) is offered, covering its methods, restrictions, and research issues [28]. **Jagdeep Singh et al.** presented a comprehensive literature review on fog computing, discussed key features of fog computing frameworks, and pinpointed the different problems with regard to its architectural design, QoS metrics, implementation specifics,

applications, and communication modalities. The article also examined the various taxonomically-based research projects and offered a classification based on the available literature for fog computing frameworks [29]. **Christos L. and Stergiou et al.** emphasized that 6G is a new sort of network architecture that yields all the advantages of its predecessors while simultaneously overcoming their drawbacks. Taking into consideration that telecommunications-related technologies like Cloud Computing, Edge Computing, and IoT can function on a 6G network, the author proposes a scenario that attempts to incorporate IoT functions with Cloud Computing, BigData, and Edge Computing to attain an intelligent and safe environment. Furthermore, the study presents a new and safe Cache Decision System in a wireless network that functions on a Smart Building, providing viewers with a secure and productive surrounding for surfing the internet, communicating, and maintaining big data in the fog [30].

Hazra, A., Adhikari, et al., developed a 6G-aware fog federation model for incorporating optimal fog needs and ensuring requirement-specific services all over the network while optimizing fog network operator profits and assuring the least delay in the service and cost for IoT users. A non-cooperative Stackelberg game connectivity method has been established to distribute fog and cloud resources by improving dynamic service expenditure and client requests. A resource console is activated to handle accessible fog assets, generate revenue for service suppliers, and guarantee the effortless quality of support. A comprehensive simulation study of 6G-aware performance specifications shows the advantage of the proposed model, which restricts latency to 15–20% and customer service to 20–25% when contrasted with standalone cloud and fog paradigms [31].

U.M. Malik et al. focus on reviewing the technologies that enable massive IoT and 6G. The authors also explore the energy-related complexities of fog computing in large-scale IoT enabled by 6G networks. In addition, they classified various energy-efficient fog software and services for IoT and characterized recent work in each of these segments. Subsequently, the authors analyze potential prospects and research issues in establishing energy-efficient fog technology for the upcoming 6G massive IoT networks [32]. The 6G enabled Network in Box (NIB) technology is demonstrated by **Baofeng Ji et al.** as a robust integrated platform capable of supporting complete network operation and management. This 6G-based NIB may be utilized as a substitute to address the demands of next-generation cellular networking by reconfiguring the network functionality deployment dynamically, giving a high level of elasticity for communication services in various scenarios. In particular, the computational intelligence (CI) technology used as part of NIB, including neural computing, evolutionary computing (EC), and fuzzy systems (FS), has implicit abilities to manage various uncertainties, offering exceptional benefits in processing the discrepancies and diversification of large datasets [33]. According to **Asif Ali Laghari et al.**, the things in the IoT are similar to humans and computers in that they can be allocated IP (internet protocol) addresses and transmit data across networks or some other man-made thing. In this work, authors explore the use of IoT and empowering technological advancements, including cloud, fog, and 6G, in conjunction with sensors, applications, and security. Researchers discovered that

sensors are critical elements of the IoT environment; if sensors fail while observing the working atmosphere, operating a vehicle, or in healthcare applications, a substantial loss will occur [34]. **Mung Chiang and Tao Zhang** highlighted the fog opportunities and limitations, concentrating particularly on the IoT networking aspect. This study defines fog as an emerging architecture for processing, storage, management, and networking that brings cloud-to-things services proximate to end-users. Fog as an architecture enables an increasing number of applications, such as the IoT, embedded artificial intelligence (AI), and Fifth Generation (5G) wireless systems. This article also explores why a new architecture called “Fog” is needed and how it could bridge technological shortfalls and provide new economic opportunities [35].

3 Overview of 6G Network, IoT, and Fog Computing

3.1 6G Vision

In 1926, Nikola Tesla stated: “*The entire planet would be transformed into a giant brain when wireless is appropriately deployed.*” Following the ideology of this visionary giant, a new vision of a 6G network has been proposed. The world is already witnessing how lifestyles and industries are progressively becoming data-driven and autonomous. This trend is anticipated to accelerate in the coming years. Various international efforts are being made by dominant countries in the wireless network industry for relevant B5G (Beyond 5G) as well as 6G initiatives and investments. Aside from that, a number of nations have started their own initiatives and given funds to conduct their studies. China, South Korea, Japan, Finland, Australia, the United States, and the United Kingdom are all in the running, and other countries are under pressure to join as well. Examples of ongoing Beyond fifth-generation (B5G) and 6G initiatives have been illustrated in Fig. 2 [36]. The next technological revolution is being driven by the merging of the digital and physical worlds, as well as automation and networked intelligence assistance. The line separating artificial intelligence, computer science, and telecommunications is blurring, allowing for a slew of new applications but also posing a challenge to future 6G networks in terms of complexity and cost to deliver additional services. The ITU (International Telecommunication Union) 2030 organization presented the first conjectural perspective on upcoming 6G services as well as use cases, emphasizing VR (virtual reality) and MR (mixed reality) services as key drivers for future 6G services.

Fundamental Enabling Technologies This study predicts five key technological necessities that will be required to meet the demands of the B5G/6G system and realize the paradigm transition from the IoT to the Internet of Intelligence, with the latter defined as functions that can represent information, process knowledge, and make choices [37].








Country	B5G/6G gambit
	- "Secure 5G & Beyond Act" March 2020 - DoD Testbed programme, US\$ 600 million - Next-G initiative, industry federation
	- Australian Digital Economy Strategy, AU\$ 1.2 billion - Modern Manufacturing Initiative, AU\$ 1.3 billion - 5G & 6G Security and Testbed, AU\$ 31.7 million / 4 years
	- MSIT 6G programme, September 2020 - US\$ 200 million public support
	- MIC "Roadmap towards 6G", June 2020 - METI Support - US\$ 380 million
	- 6G Smart Network and Services Joint Undertaking proposal - € 900 million / 7 years
	- 6G Flagship launched in February 2019 - € 250 million / 7 years
	- MIIT 6G programme, creation of IMT 2030 Promotion committee (2019) - Multi € billion until 2035, including industrialization

Fig. 2 Global examples B5G/6G initiatives

- Artificial Intelligence:** The first paradigm transition is from a traditional 5G system and its upcoming releases, i.e., an AI-enhanced network, to an AI-native communication platform [37]. An AI-native 6G technology might provide conceptual communication functionality out of the box, mimicking how the human brain works. The widespread deployment of DNN (deep neural networks), which enables usable and comprehensible meanings to be generated from an infinite quantity of processed data, may facilitate semantic and goal-oriented communication [38]. Moreover, the design, control, and administration of next-generation wireless networks can be greatly aided through the widespread use of GTP (generative pre-trained transformer platforms) [39].
- Combined Sensing and Communication:** The next paradigm transition is to uplink and downlink sensing from an information-centric approach of bits and bytes (Fig. 3), with sensors infused in access points (radio heads) and devices, referred to as Neural Edges, that operate at exceptionally high frequencies lying in the spectrum of millimeter waves (mmW) and terahertz (THz) and utilizing extremely wide detachable and contiguous bandwidths (of beaucoup GHz). Sensing is a key aspect of future 6G networks and gadgets since it is the most



Fig. 3 From data-centric to sensor-and-communication-integrated

fundamental form of intelligence. Aside from the traditional network quality indicators and wireless resource measures, it is expected to have a complete ability to detect the surroundings and aspects, similar to modern lidar or radar systems, and so retrieve a large amount of data. Sensing as a Service may be offered by blending this data with other data sources, photos, or anything else that could be acquired by any other sensors [37].

3. **Connectivity in Air, Space, and Extreme Lands:** The future generation of communication networks is projected to deliver ubiquitous services in previously unreachable places, such as outer space and across vast oceans. Terrestrial (land-based and marine), aerial (balloons, airplanes, pseudo satellites, drones, etc.), and space-based (LEO: Low Earth Orbiting, MEO: Medium Earth Orbiting, GEO: Geo-stationary Earth Orbiting) satellite constellation infrastructures will be combined to provide a seamlessly integrated connectivity architecture.

NTN (Non-Terrestrial Networks) is defined as a network in which airborne (i.e., UAS (Unmanned Aerial Systems) or space-borne (LEO, MEO, GEO) and HAPS (High Altitude Platform Systems) automobiles act as base stations (BS) or relay nodes; infrastructure will be an essential component to access network of 6G services and backhaul of upcoming generation Information and Communication Systems (ICS) and will accomplish the requirements of service availability, reliability, and consistency over broad areas. The capacity of NTNs to provide wide-area coverage by offering connections across places (e.g., airplanes, rural areas, and submarines) that are exorbitant or impossible to access in terrestrial networks is its distinguishing feature. As a result, the NTN represents a terrestrial network coverage expansion in a global industry with high sales for various services continually expanding due to an escalating number of interconnected devices

4. **Security Controls and Assurance, and Privacy Preservation:** The fourth paradigm transition concerns privacy protection and cyber security generally: wireless 6G is anticipated to be secure in terms of design rather than being a security-enhanced system, as 5G is today when compared with 4G [40]. Though it is arduous to envisage preemptive security controls because wirelesses 6G has yet to be specified and agreed upon by any SDO (standards development



Fig. 4 From security-enhanced networks to security-by-design infrastructures



Fig. 5 From operator-centric to prosumer-centric systems

organization), it is critical to recognize that by investigating the exposure to the risk of the projected 6G networks thoroughly, a tentative evaluation of potential attacks may be undertaken. New threats will emerge as a result of technological advancements; these must be addressed alongside any current threats that are being transferred from networks of previous generations [41]. To summarize, 6G must incorporate security into the infrastructure’s core and embed a defense-in-depth approach across the network, complemented by a Zero-Trust model [16], with the capacity to cope with various scenarios and unforeseen occurrences in harsh conditions in order to transition from a security-enhanced network to a security-by-design system (Fig. 4). In addition, new procedures for security management, security assurance, and privacy protection must be included in the 6G standardization process [37].

- 5. Prosumer Centric Systems:** The final but essential paradigm change is the transition to a truly user-oriented system from an operator-centric one, which is no more than a generic stream of bits. The user is predicted to evolve into a true prosumer, meaning that they will not only be able to consume material and information but will also be able to create and distribute content, making it accessible to diverse people’s communities and cyber authorities through the usage of 6G services (Fig. 5) [37].

6G Services 6G services are expected to be available beginning in 2030 and lasting for the next 10–15 years. Several of these services will be available first with 5G technologies or their long-term progression; others would necessitate technological

advancement and revolutionary network functions to fulfill their rigorous criteria as part of their typical paradigm of inexorable technological progress. In researchers' opinion, in addition to the services now offered by 5G networks, 6G will include the following services to satisfy the multiplicity of new use cases:

1. **Humongous Machine-Type Communication for Decentralized Intelligence (HMTCDI) service:** The Machine-Type Communications (MTC) of Future 6G will expand the functionality of immense MTC envisaged in 5G to incorporate ubiquitously decentralized compute techniques that support the 6G network's distributed intelligence, continuing the paradigm change started with 5G. This new service will be defined by its criticality, scalability, and effectiveness. Super smart cities, intelligent transportation systems, connected living, and other applications will be familiar.
2. **Globally-Advanced BroadBand (GABB) service:** This service is well-known for extending the computation-oriented communication system to accommodate rate-hungry tasks, including remote regions, such as oceans, the skies [42], and the rural areas [16], or the extended reality services, on-demand, at what time or which place.
3. **Ultra-Reliable, Low Latency Communication, Processing, and Control (URLLCPC) service:** These services are regarded to expand the capabilities of URLLCP services, which are presently 5G network supported, by incorporating compute services at the network's edge as well as E2E (End-to-end; remote or automated) control. Latency and reliability within the particular service are components of communication and the computation side, such as accurate classification probability or learning accuracy. The service will support factory automation, networked and autonomous terrestrial, multi-sensor XR (Extended Reality) [43] and aerial vehicles, and other use cases.
4. **Semantic Service:** All applications that involve the sharing of knowledge between the parties that interact will be supported by these services. The limit is not only set for H2H (human to human) interactions, but the application will also offer M2M (machine to machine) and H2M (human to machine) communication. The intertwining and seamless connectivity of many types of intellect, both artificial and natural, will be the service's goal. Affective computing, bi-directional and autonomous collaboration among distinct CPS (Cyber-Physical-Spaces), empathic and haptic communication, and other technologies will be enabled. This new service will be the first to offer an intelligence as a service, ushering in a profound transition that will completely transform wireless telecommunications from linked devices to linked intelligence.

6G Use Cases 6G networks entail use cases that were proclaimed throughout 5G but not yet attained, and also more sophisticated dilemmas emanating from the perspective of future generation/6G, which include haptic/tactile communications, omnipresent services (land, sea, air, space), healthcare services, national/government security, and so on. The following are some examples of related use

Table 1 6G use cases, related technological requirements, and services

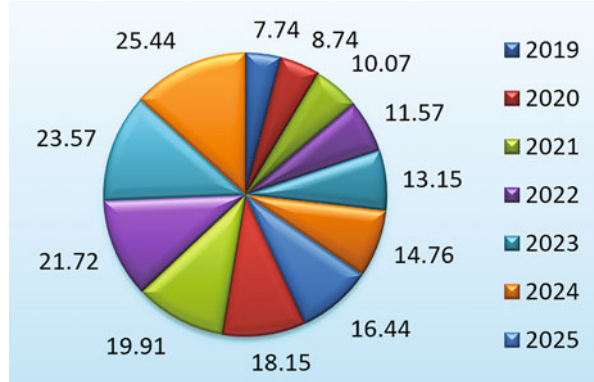
Technology requirement	Use cases	Service
Enhanced reliability	Digital twin	1,2,3,4
	Transportation	1,2,3
	Haptic/Tactile communications	3,4
	Healthcare	1,3
	Tele-surgery	—
	National/Government security	—
	Emergency services/first responders	—
Very high bandwidth	Holographic	3,4
	Tactile/Haptic communications	3,4
	Digital Twin	1,2,3,4
Synchronization of multiple flows to multiple devices	Digital Twin	1,2,3,4
	Holographic communications	3,4
	Haptic/Tactile communications	3,4
	Virtual healthcare services	1,3
	Tele-surgery	—
	Omnipresent services	—
Very wide coverage	Monumental scale IoT networks	—
	Transportation	1,2,3
	Smart farming and livestock	1,2,3
	Tactile/Haptic communications	3,4
Precise position tracking	Transportation vertical	1,2,3
	Massive scale of lot networks	—
Extremely low power and resource-constrained devices	Smart agriculture and livestock	1,2,3

cases, technical requirements, and associated services (shown in Table 1; where 1. MMTCCxDI, 2. GeMBB, 3. URLLCC, 4. Semantic) [44].

3.2 Fog Computing

A cascade amount of data is produced as the number of IoT devices, mobile Internet, and other networked items increases [45]. According to a report by Statista, it is estimated the world will witness an upsurge in the number of IoT devices to triple by 2030, i.e., from 8.74 billion in 2020 to more than 25.4 billion (Fig. 6). As a result, conventional computing architectures, including distributed and cloud computing, appear insufficient to handle such massive data [46]. Some applications, along with smart healthcare emergency response, traffic signal infrastructures, as well as smart grids, need rapid reaction and mobility assistance [47]. To solve

Fig. 6 IoT devices worldwide (in billions)



these difficulties, which include privacy-sensitive applications, ultra-high latency, large bandwidth, and geographic distribution, a computational architecture that facilitates cloud technology and executes the requested tasks of IoT devices with the shortest turn-around time and the lowest latency is necessary [48]. Cisco launched fog computing in 2012 to mitigate issues with IoT devices in the standard cloud technology [49]. Edge computing, i.e., a subdivision of FC, allows edge devices to execute computing and storage tasks hither to the edge. The massive IoT enabled by 6G will rely heavily on fog computing. With so many devices exchanging data and running several apps, the fog nodes' computing and storage support for the end devices will be essential. Furthermore, these fog nodes will be heavily used in 6G communications to allow ubiquitous connection and achieve exceptionally low latency. This new fog computing-based 6 G-enabled large IoT paradigm will address several issues in terms of allocating resources; energy consumption, fairness, work offloading, reduced latency, and security.

3.3 Massive IoT

The term “massive IoT” is used to describe the large-scale connectivity of a massive number of devices, sensors, and machines [16, 42]. IoT intends to link diverse items to the Internet, generating a linked world in which data collection, computing, and communication are carried out autonomously and without user intervention. It is a fundamental technique for interconnecting diver-gent electronic equipment with wireless connections. For the service of end users, IoT data may be acquired through prevalent digital assistants, including computer systems, sensor devices, mobile phones, actuators, computer systems, and radio frequency identification (RFID) [43]. The IoT is expected to grow rapidly in the upcoming years. Cisco [50] estimates that approximately 0.5 trillion IoT devices will be linked in 2030 to the World wide web, up from 26 billion in 2020. According to GlobeNewswire, the 5G-IoT market is expected to expand to 6.2855 billion

dollars in 2025 [51]. The Internet of Nano-Things is similarly important for the development of future advanced IoT ecosystems in which a network of items (nano-devices and things) may detect, process, transfer, and store data using nanoscale elements (e.g., a nano controller) to support client operations like patient monitoring [52]. Interconnecting nano-networks through accessible internet and communication networks in a seamless manner necessitates the development of new network topologies and communication paradigms. In this regard, 6G will be a key facilitator towards upcoming IoT platforms and devices since it will deliver multi-dimensional wireless connectivity and consolidate all functions, such as sensory, communication, processing, intelligence, and completely autonomous operation. In reality, the upcoming generation (6G) wireless networks are projected to provide greater coverage and flexibility than the 5G networks, enabling IoT connection and service provision [53].

IoT network density is extremely high, attaining over 1 million units per square kilometer [54]. Consequently, tremendous amounts of data will need to be spread across devices, a significant number of activities will need to be completed, and massive amounts of data will need to be stored and big data evaluated. Consequently, fog computing and 6G will be critical enablers for large-scale IoT applications. Several features are responsible for attaining intelligent 6 G-enabled IoT, some of which are discussed below:

1. **Latency Reduction:** Reducing latency results in the high-speed execution of jobs that enhance a service's experience for users, which would be critical for its continuation in a fast-paced industry.
2. **Fair Offloading:** Fair offloading among gadgets while retaining adequate energy efficiency is crucial, not just for the network's long-term viability but also for the owner's contentment in continuing to share resources. Conversely, fairness is accomplished at the expense of extended delays. Therefore, to assure job completion within a given time frame, a trade-off between delay and fairness is required.
3. **Load Balancing:** It is a critical QoS (Quality of Service) element that ensures no device is overused and network operations are stabilized. In general, load balancing minimizes job queues, which reduces task execution time.
4. **Resource Allocation:** Every accessible computation node/device, as well as associated resources and data regarding their current computational jobs, must be examined by resource managers to allocate resources efficiently. These devices must balance multiple calculations, communication, and latency limitations to disperse the workload to achieve energy efficiency without overburdening the computing devices and other resources. In fog technology, IoT networks, precise system knowledge, and learning techniques aid in allocating resources.
5. **Fault Tolerance:** It refers to the system's capacity to provide the necessary service not with-standing specific system faults. Fog nodes have built-in fault tolerance to ensure that all assigned tasks are completed. Offloaded tasks are monitored, and if a fog node exits the system due to a power outage and leaves a job incomplete, that work is offloaded to another fog node for completion.

4 Fog Enabled Intelligent IoT Applications: Trends and Challenges

FC is a viable technique for upcoming 6G networks capable of providing computational and storage capabilities. This technology will be critical in supporting 6 G's enormous IoT applications since IoT and fog node devices have limited energy and computing solutions that necessitate intelligent energy-efficient storage. An overview of fog-based massive IoT and 6 G-enabled IoT-based applications is presented. There are numerous AI applications that make use of fog IoT networks; a few of these are discussed below [55–57]:

1. **Wireless Communication Network:** Fog would enable self-optimized processing, memory, management, and connectivity capabilities to move flexibly between the cloud, devices, network edge, and fog. Advanced and sophisticated gadgets like phones, laptops, and tablets are straining conventional wireless networks to their constraints. With a synergy of various Radio Access Technologies (RATs) such as 5G new radio (NR), LTE/LTE-Advanced, Internet-of-Things, Wi-Fi, 6G, and others, upcoming wireless communication infrastructures are projected to be intensively distributed and versatile in dealing with the constant traffic growth. Fog can access edge nodes and enterprise clients, which enables fog computing to take advantage of network edge processing applications, coordinated resource allocation, and distributed storage features. When a wireless system is fog-enabled, a significant portion of signal analysis and computation is dispersed, and regional information can be retained and analyzed at the network edge and user devices, facilitating programs that require deficient mobility and latency [58]. Offloading computationally complex operations to the fog node adjacent to the software platform can, for instance, significantly minimize application execution latency.
2. **Intelligent Transportation System:** As people's reliance on transport services expands, transport systems confront various issues, including traffic jams, accident rates, and effective transport organization. Utilizing data, connectivity, control, computerization, and other contemporary advancements, an Intelligent Transportation System (ITS) is intended to build a real-time, precise, and effective system for transportation. An ITS maximizes the system's effectiveness in terms of the flow of traffic, reliability, latency, and energy consumption by integrating data from a range of sources, such as sensors, navigation systems, and other vehicles. Fog computing is an important paradigm in modern network-connected society as it allows for low delay, high durability, and 24/7 service for applications [56]. It facilitates essential ITS operations by communicating, coordinating, and exploiting the underlying network capabilities within roads, smart cities, and highways. Fog will aid the adoption of many private and commercial autonomous vehicles by eliminating the issues associated with ITS.
3. **Collaborative Robot System:** Simultaneous localization and mapping (SLAM) is the synchronous construction of a surrounding map and assessment of the

machine's status in robotics [59]. Minimal cost, power-efficient, precise, and quick robot SLAM is essential, although such needs impose reciprocal constraints. To obtain precise mapping and positioning, a powerful computational unit is required, particularly in the optimization phase, which comprises various sophisticated analytics, and it is certainly contrary to the minimal-cost criterion. Using relatively moderate algorithms is an appropriate strategy to conserve the robot's power requirements; moreover, this can lead to inconsistent SLAM. Furthermore, in several rescue instances, timing is crucial; as a result, the robot must act rapidly and execute quick SLAM, imposing additional strain on the inbuilt computer unit. SLAM speed can also be increased by employing low-complexity algorithms. However, there is a phase in which SLAM precision may be compromised. In a broad region with several robots, the robots need to engage in SLAM and eventually unify the maps, building a network and designating a robot as the leader in merging the SLAM and maps.

4. **Smart Home:** A smart home is just a distinct IoT system in which all digital equipment can be linked to the Internet and execute certain computer functions. Each device can be considered an IoT node and constitutes a local network. 6G empowered fog provides a ray of light for smart home implementation. Fog's multilayered structure allows the inclusion of its own fog nodes on each level or node to develop a control system with a hierarchical structure. Every fog node may be liable to perform emergency tracking and response operations, construct protection capabilities, and control the temperature and lighting of a home [56]. Also, they may offer a better storage and computation platform for citizens to guide sensors, computers, mobile phones, etc. Local sensors can transfer sensory or tracking data to nearby fog nodes first. The information will be preprocessed by fog nodes, making analysis easier.
5. **Smart Cities:** Another potential application of 6G-IoT is smart cities. Some examples of IoT applications for smart metropolitan cities include autonomous transportation, urban security, intelligent energy management systems, intelligent surveillance, water supply, and environmental monitoring. It is well known that several Smart-city strategies assert to enhance the daily life of citizens living in urban communities [57]. Smart city IoT technologies alleviate traffic congestion, minimize noise pollution, and aid in the security of metropolitan areas.
6. **Smart Retail:** In the field of Smart Retail, IoT fueled with fog and 6G would operate excellently. Retailers could interact with their clients more rapidly and conveniently with this support. The mobile phone would be the most common tool for this task. Clients might even purchase through their mobile phones using a mobile payment system, and they can track their orders.
7. **Digital Health:** IoT in healthcare has gained prominence in recent years and will continue to advance in the future. IoT in healthcare encourages individuals to use smart devices to live a healthy lifestyle. Although the notion of connected digital healthcare has enormous potential for both individuals as well as the medical and pharmaceutical sectors, it has yet to reach the vast majority of the population.

8. **Smart Farming and Environment:** The need for food is growing as the world's population grows. In such a context, smart farming is among the most rapidly and crucially expanding fields of fog-enabled IoT. It not only assists farmers or agricultural enterprises to earn more profit but also allows consumers to acquire food at a lower cost and of higher quality. Farmers are utilizing fog-IoT-enabled devices to regulate vegetation water supply and gather intelligence on soil nutrients and moisture. Farmers in the field use sensors to evaluate natural boundaries (such as temperature and humidity), and this data could be used to increase the efficiency of production. One example is a robotized water system that responds to weather conditions. Natural boundaries are observed gradually in terms of temperature, soil nutrients, as well as moisture and forwarded to the fog servers for assessment. Obtained results can then be utilized to enhance the product's quality and increase production. Air pollution is a problem nowadays, affecting the ecological climate and degrading air quality. IoT software powered by 6G-fog monitors vehicles that can cause an excessive level of contamination. Electrochemical toxic gasoline sensors can also be used to quantify air pollution. RFID (Radio-frequency identification) stickers make vehicles stand out. On the two roads, RFID readers are installed together with gas sensors. With this technique, it is significantly more feasible to distinguish and take action against polluted automobiles [34].
9. **Society 5.0:** Professor Harayama created the concept of "Society 5.0," claiming that it attempts to address numerous modern societal concerns by integrating game-changing technologies, including IoT, automation, big data, and AI, into all sectors and social activities [57]. Instead of a future operated and supervised by Robotics and AI, Technologies are being utilized to establish a human-centered future where everyone will live an active and joyful life. Having tech, nature, and social systems operating in a balanced scope, one can keep a building efficient, supplying energy to a smart city and ensuring that all services provided by that city are efficient and available. This is made possible by a high level of integration between cyberspace (virtual space) and real space (physical space). In Society 5.0, a large volume of data collected through sensors in real space is collected and stored in virtual space. If society 5.0 is empowered with fog and 6G technology, it will be capable of enhancing its performance and transporting information more quickly and securely [60].
10. **Industry 4.0:** Industry 4.0 (I4.0) is intended to provide the manufacturing industry with new opportunities, such as satisfying customers' requirements, maximizing decision-making, and adding additional application capabilities [57]. The I4.0 reformation is viewed as the amalgamation of two worlds: (1) physical (robotic and automation systems) and (2) virtual (big data and AI), to build the concept of smart factories via the IoT. This advancement has enabled several innovations, such as cooperative robotics as well as quality control through digital channels, and sensors have the potential to increase efficiency by 45–55%. Sensory technological advances have the prospect of obtaining vast volumes of data from complex applications. AI, cloud, fog, and IoT are expected

Table 2 IoT challenges and their possible solution

IoT challenges	The advantages of Fog to overcome IoT challenges
Resource-constrained devices	When resource-constrained devices cannot be relocated to the cloud server for some reason, Fog could perform resource-intensive operations on their behalf, decreasing the complexity, lifespan costs, and energy consumption of these devices.
Latency constraints	To accomplish the rigorous timing constraints among several IoT systems, fog, which performs data processing, management, and many other time-sensitive functions in proximity to end users, is the optimal and sometimes the only alternative.
Uninterrupted services with intermittent cloud connectivity	A local Fog system can perform independently to deliver uninterrupted services, including while network access to the cloud is sporadic.
Network bandwidth constraints	Fog supports multidimensional data computation throughout the Cloud-to-Things continuum, enabling computation to take place where it can stabilize application necessities with accessible connectivity and computational resources. This decreases the volume of data that must be transmitted to the cloud.
IoT security challenges	For example, a fog system can (1) serve as proxies to resource-constrained systems to assist in maintaining and upgrading security credentials as well as software; (2) track the security state of connected devices; (3) execute a multitude of security activities, including malware scans, for resource-constrained systems to accommodate for the limited security proficiency; and (4) make use of location data and environment to identify threats on a telecommunications network, in real-time.

to have the most significant effect, while edge computing, quantum computing, blockchain, and 3D printing are expected to have the least impact [61].

5 Fog as a Solution to IoT Challenges

Several drawbacks of present computer infrastructures that depend solely on cloud computing and end-user devices can be addressed by adopting fog. The table below (Table 2) illustrates how Fog might assist in addressing IoT challenges [35, 56, 57].

6 Conclusions and Summary

IoT technology is anticipated to deliver new service models in several domains, including smart cities, smart grids, healthcare, smart transportation, rural area coverage, and more other services to facilitate faster as well as highly secure data processing for IoT users. All these IoT applications require ultra-fast connections (i.e., B5G/6G technique) as well as collaborative fog computing characteristics for

their effective functioning. Fog computing with 6G communication technologies, if developed successfully, might unveil novel avenues for network administrators, cloud vendors, and diverse IoT users, allowing numerous network administrators and service providers to work together to manage the users' demands. It facilitates excellent services to IoT customers, and they can enjoy high QoS factors, including fast data speed, uninterrupted internet connectivity, interoperability, and continuous innovation, resulting in a rise in the ratio of user satisfaction. To enhance the network performance, idle and spare resources, including all accessible devices, will be intelligently integrated through fog devices, as fog computing acts as a critical component in envisioned 6G technologies. This study summarizes the aforementioned technologies to illuminate concepts, including various fog-enabled IoT applications in B5G/6G networks, along with several challenges that IoT may encounter and provides possible fog solutions for the same. In essence, the research aimed to present a study to investigate the recent contributions to scientific studies on fog computing and IoT in the 6G-enabled contemporary age and to indicate prospective study and open challenges engulfing the merging of intelligent fog-IoT networks with 6G.

References

1. *How the world will change as computers spread into everyday objects*. Accessed May 25, 2022. Available online: <https://www.economist.com/leaders/2019/09/12/how-the-world-will-change-as-computers-spread-into-everyday-objects>
2. Pham, Q.-V., Fang, F., Ha, V. N., Le, M., Ding, Z., Le, L. B., & Hwang, W.-J. (2020). A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art. *IEEE Access*, 8, 116974–117017.
3. Basel, B., Ahmad, T., Samson, R., Steponenaite, A., Ansari, S., Langdon, P. M., Wassel, I. J., Abbasi, Q. H., Imran, M. A., & Keates, S. (2021). 6G opportunities arising from internet of things use cases: A review paper. *Future Internet*, 13(6), 159.
4. Chiang, M., & Zhang, T. (2016). Fog and IoT: An overview of research opportunities. *IEEE Internet of Things Journals*, 3(6), 854–864.
5. Ahmad, A., Abdullah, S., Iftikhar, S., Ahamd, I., Ajmal, S., & Hussain, Q. (2022). A novel blockchain based secured and QoS aware IoT vehicular network in edge cloud computing. *IEEE Access*, 10, 77707–77722.
6. Lin, C., Han, G., Qi, X., Guizani, M., & Shu, L. (2020). A distributed mobile fog computing scheme for mobile delay-sensitive applications in SDN-enabled vehicular networks. *IEEE Transactions on Vehicular Technology*, 69(5), 5481–5493.
7. Mukherjee, M., Kumar, S., Mavromoustakis, C. X., Mastorakis, G., Matam, R., Kumar, V., & Zhang, Q. (2019). Latency-driven parallel task data offloading in fog computing networks for industrial applications. *IEEE Transactions on Industrial Informatics*, 16(9), 6050–6058.
8. Luo, S., Chen, X., Zhou, Z., Chen, X., & Wu, W. (2020). Incentive-aware micro computing cluster formation for cooperative fog computing. *IEEE Transactions on Wireless Communications*, 19(4), 2643–2657.
9. Tange, K., Michele, De. D., Fafoutis, X., & Dragoni, N. (2020). A systematic survey of industrial internet of things security: Requirements and fog computing opportunities. *IEEE Communications Surveys & Tutorials*, 22(4), 2489–2520.

10. Adhikari, M., Mukherjee, M., & Srirama, S. N. (2019). DPTO: A deadline and priority-aware task offloading in fog computing framework leveraging multilevel feedback queueing. *IEEE Internet of Things Journal*, 7(7), 5773–5782.
11. Aazam, M., Zeadally, S., & Harras, K. A. (2018). Deploying fog computing in industrial internet of things and industry 4.0. *IEEE Transactions on Industrial Informatics*, 14(10), 4674–4682.
12. Ansar, S. A., Arya, S., Aggrawal, S., Yadav, J., & Pathak, P. C. (2022). Bitcoin-blockchain technology: Security perspective. In *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*. IEEE.
13. Ali, S., Sohail, M., Shah, S. B. H., Koundal, D., Hassan, M. A., Abdollahi, A., & Khan, I. U. (2021). New trends and advancement in next generation mobile wireless communication (6G): A survey. *Wireless Communications and Mobile Computing*, 2021, 14 pp.
14. Sodhro, G. H., Zahid, N., & Rawat, A. (2019). A novel energy optimization approach for artificial intelligence-enabled massive internet of things. In *2019 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*. IEEE.
15. Stutek, M., Zeman, K., Masek, P., Sedova, J., & Hosek, J. (2019). IoT protocols for low-power massive IoT: A communication perspective. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. IEEE.
16. Lee, G., & Youn, J. (2020). Group-based transmission scheduling scheme for building LoRa-based massive IoT. In *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*. IEEE.
17. Zeadally, S., Javed, M. A., & Hamida, E. B. (2020). Vehicular communications for its: Standardization and challenges. *IEEE Communications Standards Magazine*, 4(1), 11–17.
18. Jameel, F., Javed, M. A., & Ngo, D. T. (2019). Performance analysis of cooperative v2v and v2i communications under correlated fading. *IEEE Transactions on Intelligent Transportation Systems*, 21(8), 3476–3484.
19. Javed, M. A., & Zeadally, S. (2018). Repguide: Reputation-based route guidance using internet of vehicles. *IEEE Communications Standards Magazine*, 2(4), 81–87.
20. Giordani, M., Polese, M., Mezzavilla, M., Rangan, S., & Zorzi, M. (2020). Toward 6g networks: Use cases and technologies. *IEEE Communications Magazine*, 58(3), 55–61.
21. Saad, W., Bennis, M., & Chen, M. (2019). A vision of 6g wireless systems: Applications, trends, technologies, and open research problems. *IEEE Network*, 34(3), 134–142.
22. Mao, B., Kawamoto, Y., & Kato, N. (2020). AI-based joint optimization of QoS and security for 6g energy harvesting Internet of Things. *IEEE Internet of Things Journal*, 7(8), 7032–7042.
23. Sodhro, A. H., Sodhro, G. H., Guizani, M., Pirbhulal, S., & Boukerche, A. (2020). AI-enabled reliable channel modeling architecture for fog computing vehicular networks. *IEEE Wireless Communications*, 27(2), 14–21.
24. Luong, N. C., Jiao, Y., Wang, P., Niyato, D., Kim, D. I., & Han, Z. (2020). A machine-learning-based auction for resource trading in fog computing. *IEEE Communications Magazine*, 58(3), 82–88.
25. Lin, C., Han, G., Qi, X., Guizani, M., & Shu, L. (2020). A distributed mobile fog computing scheme for mobile delay-sensitive applications in SDN-enabled vehicular networks. *IEEE Transactions on Vehicular Technology*, 69(5), 5481–5493.
26. Rahim, M., Javed, M. A., Alvi, A. N., & Imran, M. (2020). An efficient caching policy for content retrieval in autonomous connected vehicles. *Transportation Research Part A: Policy and Practice*, 140, 142–152.
27. Javed, M. A., Nafi, N. S., Basheer, S., Bivi, M. A., & Bashir, A. K. (2019). Fog-assisted cooperative protocol for traffic message transmission in vehicular networks. *IEEE Access*, 7, 166148–166156.
28. Chakraborty, A., Kumar, M., Chaurasia, N., & Gill, S. S. (2022). Journey from cloud of things to fog of things: Survey, new trends, and research directions. *Software: Practice and Experience*, 53, 496–551.

29. Singh, J., Singh, P., & Gill, S. S. (2021). Fog computing: A taxonomy, systematic review, current trends and research challenges. *Journal of Parallel and Distributed Computing*, 157, 56–85.
30. Soldani, D. (2020). On Australia's Cyber and Critical Technology International Engagement Strategy towards 6G – How Australia may become a leader in Cyberspace. *Journal of Telecommunications and the Digital Economy*, 8(4), 127–158.
31. Stergiou, C. L., Psannis, K. E., & Gupta, B. B. (2020). IoT-based big data secure management in the fog over a 6G wireless network. *IEEE Internet of Things Journal*, 8(7), 5164–5171.
32. Hazra, A., Adhikari, M., Amgoth, T., & Srirama, S. N. (2020). Stackelberg game for service deployment of IoT-enabled applications in 6G-aware fog networks. *IEEE Internet of Things Journals*, 8(7), 5185–5193.
33. Malik, U. M., Javed, M. A., Zeadally, S., & UI Islam, S. (2021). Energy efficient fog computing for 6G enabled massive IoT: Recent trends and future opportunities. *IEEE Internet of Things Journal*, 9, 14572–14594.
34. Ji, B., Wang, Y., Song, K., Li, C., Wen, H., Menon, & V. G., Mumtaz, S. (2021). A survey of computational intelligence for 6G: Key technologies, applications and trends. *IEEE Transactions on Industrial Informatics*, 17(10), 7145–7154.
35. Laghari, A. A., Wu, K., Laghari, R. A., Ali, M., & Khan, A. A. (2021). A review and state of art of Internet of Things (IoT). *Archives of Computational Methods in Engineering*, 1–19.
36. *6G Gains momentum with initiatives launched across the world*. Accessed in 2022. Available online: <https://www.6gworld.com/exclusives/6g-gains-momentum-with-initiatives-launched-across-the-world>
37. *5G, 5.5G and 6G fundamentals*. Accessed in 2021. Available Online: <https://youtu.be/2jfglScLDgw>
38. Dey, S., & Mukherjee, A. (2016). Robotic SLAM. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing Networking and Services (MOBIQ-UITOUS)*
39. Strinati, E. C., & Barbarossa, S. (2021). Beyond Shannon towards semantic and goal-oriented communications. *Computer Networks*, 190, 107930.
40. Ansar, S. A., & Khan, R. A. (2018). *Networking communication and data knowledge engineering* (pp. 15–25). Singapore: Springer.
41. *Security considerations for the 5G era*. Accessed in 2020. Available Online: <https://www.5gamericas.org/wpcontent/uploads/2020/07/Security-Considerations-for-the-5G-Era-2020-WP-Lossless.pdf>
42. Stutek, M., Zeman, K., Masek, P., Sedova, J., & Hosek, J. (2019). Iot protocols for low-power massive iot: A communication perspective. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops(ICUMT)*. IEEE.
43. Al-Jarrah, M. A., Yaseen, M. A., Al-Dweik, A., Dobre, O. A., & Alsusa, E. (2019). Decision fusion for IoT-based wireless sensor networks. *IEEE Internet of Things Journal*, 7(2), 1313–1326.
44. *Mobile Communications Beyond 2020 – The Evolution of 5G Towards Next G*. Available online: <https://www.5gamericas.org/wp-content/uploads/2020/12/Future-Networks-2020-InDesign-PDF.pdf>
45. Dabbaghjamesh, M., Moeini, A., Kavousi-Fard, A., & Jolfaei, A. (2020). Real-time monitoring and operation of microgrid using distributed cloud–fog architecture. *Journal of Parallel and Distributed Computing*, 146, 15–24.
46. Gill, S. S., Tuli, S., Xu, M., Singh, I., Singh, K. V., Lindsay, D., Smirnova, D., Singh, M., & Jain, U. (2019). Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges. *IEEE Internet of Things Journal*, 8, 100118.
47. Tuli, S., Basumatary, N., Gill, S. S., Kahani, M., Arya, R. C., Wander, G. S., & Buyya, R. (2020). Healthfog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments. *Future Generations Computer Systems*, 104, 187–200.

48. Amin, R., Kunal, S., Saha, A., Das, D., & Alamri, A. (2020). CFSec: Password based secure communication protocol in cloud-fog environment. *Journal of Parallel and Distributed Computing*, 140, 52–62.
49. Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the Internet of things. In *MCC'12 - Proceedings of the 1st ACM Mobile Cloud Computing Workshop*, August, 2012.
50. *Internet of Things 2016*. Available online: <https://www.cisco.com/c/dam/en/us/products/collateral/se/internetof-things/at-a-glance-c45731471.pdf>
51. *5G IoT Market by Connection, Radio Technology, Range, Vertical and Region - Global Forecast to 2025*. Available online: <https://www.globenewswire.com/fr/news-release/2019/04/19/1806975/0/en/Global-5G-IoT-Market-Forecast-to-2025-Market>
52. Balghusoon, A. O., & Saoucene, M. (2020). Routing protocols for wireless nanosensor networks and Internet of nano things: A comprehensive survey. *IEEE Access*, 8, 200724–200748.
53. Palattella, M. R., Dohler, M., Grieco, A., Rizzo, G., Torsner, J., Engel, T., & Ladid, L. (2016). Internet of things in the 5G era: Enablers, architecture, and business models. *IEEE Journal on Selected Areas in Communications*, 34(3), 510–527.
54. Sodhro, A. H., Obaidat, M.S., Pirbhulal, S., Sodhro, G. H., Zahid, N., & Rawat, A. (2019). A novel energy optimization approach for artificial intelligence-enabled massive internet of things. In *2019 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*. IEEE.
55. Qi, Q., Chen, X., & Ng, D. W. K. (2019). Robust beamforming for NOMA-based cellular massive IoT with SWIPT. *IEEE Transactions on Signal Processing*, 68, 211–224.
56. Iftikhar, S., et al. (2023). AI-based fog and edge computing: A systematic review, taxonomy and future directions. *Internet of Things*, 21, 100674.
57. Gill, S. S., et al. (2022). AI for next generation computing: Emerging trends and future directions. *Internet of Things*, 19, 100514.
58. Yang, Y., Luo, X., Chu, X., & Zhou, M. T. (2020). *Fog-enabled intelligent IoT systems* (pp. 39–60). New York: Springer International Publishing.
59. Bonomi, F., Milito, R., Natarajan, P., & Zhu, J. (2014). Fog computing: A platform for internet of things and analytics. In *Big data and internet of things: A roadmap for smart environments* (pp. 169–186). Cham: Springer.
60. Chiang, M., & Zhang, T. (2016). Fog and IoT: An overview of research opportunities. *IEEE Internet of Things Journal*, 3(6), 854–864.
61. *What Is It & How To Achieve A Human-Centered Society – IntelligentHQ*. Accessed May 25, 2020. Available online <https://www.intelligenthq.com/society-5-0-achieve-human-centered-society>

The Role of Machine Learning in the Advancement of 6G Technology: Opportunities and Challenges



Krishna Kumar Mohbey and Malika Acharya

Abstract While the world is heading towards the 5G network-oriented revolution, the implicit limitations of 5G have garnered researchers' attention. Smart IoT (Internet of Everything) services are expected to grow. 6G, with its ultra-broadband, low latency, the decentralized and intelligent network is envisioned as a possible solution even at its infancy stage. Mobile Edge computing integrated with the Internet of Things (IoT) has eased information flow but has also complicated it. AI modeling is imperative for a 6G wireless decentralized network to move to 'connected intelligence' rather than 'connected things.' Modeling, training, and decision-making on local devices would aid in the integration of network nodes. This paper presents the recent advancements in the 6G network along with the plausible limitations and challenges in massive data procurement from IoT devices over 6G. The present use cases of IoT, Artificial intelligence, and the prospective application areas of 6G are discussed and analyzed. Further, we also posit how Artificial intelligence can maneuver the processing overheads of data processing. This paper also presents the architectural nuances of 6G that would pave the path for a fast and robust network.

Keywords Internet of things · Machine learning · 6G · Artificial intelligence

1 Introduction

The advancement of IoT devices has undoubtedly changed the communication aspects of the world. Combined with wireless networks, the spatially distributed devices capable of sensing and sending signals have made communication more reliant, reasonable, and rapid. After every 10 years, the world presents itself with an upgraded version of the mobile communication network system. And as per this convention, in 2020, the world stepped into the era of a fifth generation

K. K. Mohbey (✉) · M. Acharya
Department of Computer Science, Central University of Rajasthan, Ajmer, India

(5G) network. It predominantly overcomes the limitations of fourth-generation (4G) networks with its enhanced Mobile Broadband (eMMB), massive Machine Type Communications (mMTC), and ultra-Reliable Low Latency Communication (uRLLC) [1]. It provides 20 Gbps of peak data rate, $3\times$ spectral efficiency, energy efficiency, and end-to-end latency of 1 ms. Thus, many researchers consider it the pinnacle of the communication system and strive to develop the technology to solve the complex issues of data management, security, and processing based on 5G communication. We have discussed some in the coming sections. Table 1 summarizes the abbreviations that are used in this paper for readers convenience.

Current Statistics of Wireless Communication Network But, as the data consumption and generation paradigm transforms to the machine-oriented model rather than the people-oriented model, there is a growing need for dynamic wireless communication like 5G and beyond. The expanse of wireless communication applications has tremendously increased, requiring a high capacity, ultra-dense low BER communication network. With this, the urge for a distributed heterogeneous

Table 1 Table of abbreviations

Abbreviation	Meaning
eMMB	enhanced Mobile Broadband
uRLLC	ultra-Reliable Low Latency Communication
mMTC	massive Machine Type Communications
VR/AR	Virtual Reality/Augmented Reality
CSL	Cellular Subscriber Lines
WBCI	wireless brain-computer interfaces
WMMI	Wireless mind-machine interfaces
NIN's	non-terrestrial networks
LIDAR	Light Detection and Ranging
MEC	Mobile Edge Computing
uMUB	Ubiquitous Mobile Ultra-Broadband
uHSLLC	Ultra-High-Speed with Low -Latency Communications
uHDD	Ultra-High Data Density
IoV	Internet of Vehicles
IoMTs	Internet-of-Medical-Things
IoD	Internet-of-Drones
IIoT	Industrial Internet-of-Things
IoRT	Internet-of-Remote-Things
MIMO	Multiple-Input Multiple-Output
CFO	carrier frequency offset
BIPs	breaks in the presence
ESN	Echo State Network
IoNT	Internet of Nano-Things
UAV's	Unnamed Aerial Vehicles
MTC	Machine -Type-Communication

and highly efficient 6G comes to the forefront. 6G, with its low latency, low energy consumption, high capacity and quality, and large connectivity, revolutionizes the Internet of things (IoT) infrastructure. Real-time communication becomes more complex with the large topological network of sensor-based devices and their exponential growth. As per [2], with an annual rise of 25% in smartphone users, the number will reach 80 billion by 2030. As anticipated by IoT Analytics [3], by 2025, there will be around 30 billion active IoT devices, excluding smartphones. Moreover, there will be 1854.76 billion users in the IoT market by 2028, with a compound annual growth rate of 25.4% [4]. Thus, within these 8 years, 6G and its deployment over IoT and the implicit challenges must be thoroughly analyzed, and solutions should be sought.

5G Versus 6G With the approval of the Sustainable Development Goals Agenda, smart city development has become the prime focus of the communication paradigm. A smart city deploys 6G, cloud IoT, fog computing, edge computing, and so on, increasing the need for computation-intensive yet delay-sensitive IoT-based communications [5]. The disparity between 5G and the communication requisites serve as insensitive to technological advancements. Also, the emerging trends in personal communication promote the evolution of smart vehicles that would support varied fields like healthcare, remote education, and fully autonomous driving. IoT supports these and bridges the gap between the real world and cyberspace. Leveraging these massive IoT-enabled applications over 5G is challenging and tedious. uRLLC and mMTC are inefficient in rendering these applications' coverage, localization, and prime latency requisites. Also, the large volumes of data produced at the forefront pose severe questions about the information's security, management, and privacy [6]. Thus, 5G can be considered to support vertical IoT applications like VR/AR, tactile Internet, smart vehicles, etc., But it is vulnerable to many limitations in architecture and technology. A few of the limitations are enumerated below:

- **Data rate and latency:**

It offers a peak data rate of 20 Gb/s and an experienced data rate of 100 Mb/s. However, this statistic is somewhat misleading as it relies on unrealistic assumptions and ignores the massive user population that shares this capacity. Thus, The user throughput is only 15% of the peak data rate. It provides a latency of 1 ms, but for IoT applications, a latency of less than 1 ms is required.

- **Triad of characteristics:**

As mentioned, 5G provides uRLLC, mMTC, and eMBB services. While uRLLC facilitates low-latency data transmission for small load mMTC, connect the IoT devices with small data loads which are sporadically active. With these features, 5G cannot provide adequate facilities for data-intensive, delay-sensitive,

and computation-intensive applications that might also require a combination of characteristic classes.

- **Connection and coverage capability:**

With the increase in the expanse of IoT applications, there has been an ever-increasing demand for communications across terrestrial boundaries and in space and the sea. Today's demand varies from across the mountains to the deep into the sea, to a high altitude, and even to harsh climatic zones. However, 5G and its predecessors are mostly suited for communication across terrestrial space. With the growing times, there is a need for wireless network communication even across rural areas. Rural area inhabitants lag in mobile phone services and let go of broadband services. As per [7], internet penetration is only 37% across Indian rural areas. As per the Federal communications commission (FCC), only 69% of rural America has broadband access. With the large bandwidths spectrum and Gbps speed, short and dense cellular zones exist in 5G communication. Moreover, from the profit perspective, the rural area will be allocated a 'low' band 5G; hence it would not be able to meet the requirements of such a vast number.

- **Requisites of new applications:**

5G network does not provide a robust set of features for maintaining the security and privacy of the data; hence, the adversary can easily access it, thereby putting the information's reliability, confidentiality, and authenticity at stake [8]. The research from the University of Dundee has filled the void of security characteristics from 3G to 5G, but the gap still exists with the emerging applications. This is also a key challenge for a 6G network-based application that involves a host of sensors for data procurement.

The Necessity of 6G Vision Owing to these limitations, we anticipate that 5G will reach its yield point and will be unable to support the requirements of these futuristic applications and ultimately be abated as its predecessor. Besides cost-effective solutions, it is also required that future generations cater to standalone location-based communication without consuming much battery and assuring a high data rate. Further, it will successfully cater to the 3C model of data informatics, i.e., communication, caching, and computing. Unlike 5G, which posited limited variability in 3C for edge computing [9], 6G will open new development avenues. It relies on the 4CSL model for IoT-based applications. Some factors that are the incentive for the 6G vision are discussed below.

- **10-year Convention:**

Since the inception of 1G in 1982 for commercial purposes, a new mobile generation has rolled out every 10 years. Figure 1 depicts the evolution of the cellular network. Thus, as the world stepped into 5G officially in 2020, we believe that 6G will be the next milestone to be accomplished. Hence, this span of 8 years is best suited for 6G vision.

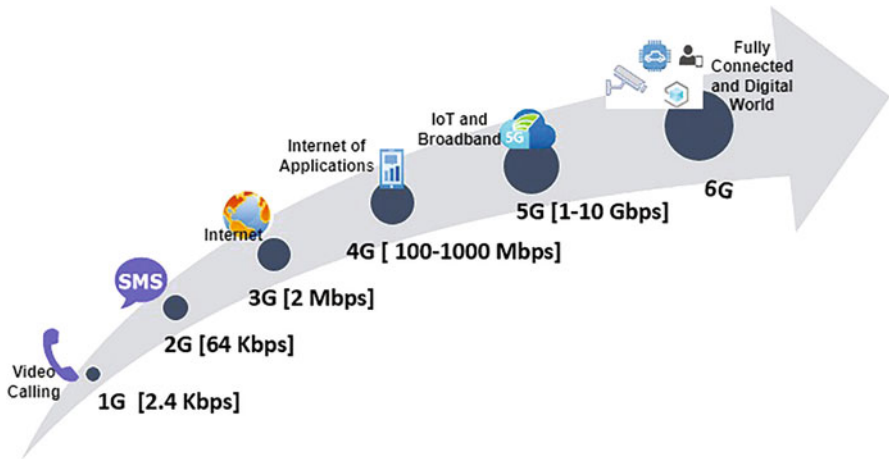


Fig. 1 Cellular network evolution

- **Catfish effect:**

There exists a catfish effect-like scenario in technology and network communication. As more new and innovative industries would participate in vertical industrialization, the 5G network meant to serve the purpose of IoT/vertical industries would eventually need to expand. This growth would lead to a gradual shift to a 6G network.

- **Emerging technological trends:**

Imagination, perspective, and planning have been the cornerstones of any technological development. The development of smartphones has rendered 3G insufficient and developed 4G. Similarly, as the technology expands, it will accelerate the growth of 6G. Before this drastic shift, there must be an imaginative perception that relies on careful and critical analysis of the short-coming and merits of 6G. Thus, 6G is imperative for future applications. Nevertheless, the shift needs to consider, analyze, and solve the impending issues like security and architectural and infrastructure requirements. With this in the backdrop, we aim to analyze the IoT based 6G communication network and the scope of Artificial Intelligence to solve the issues. With the ubiquitous localization, a vision of 6G has been framed by many researchers.

Motivation for 6G Assisted Communication Paradigm The prime motivation of the 6G empowered paradigm is rapid and scalable communication. The other incentives include:

- **Superconvergence:**

Non-3GPP wired, and radio systems are quintessential for 6G ecosystems. It is anticipated that both wireline and wireless technologies will assist in building robust

and reliable security and access control baselines for consolidated networks. The scalable convergence for different technologies will aid in traffic balancing and assist in onboarding and offloading the traffic.

- **Non-IP-based Networking Protocol:**

European Telecommunication Standards Institute (ETSI)'s Next Generation Protocol (NGP) Working Group is testing several protocols to replace internet protocols. 6G is a silver lining in wireless edge as it caters to almost 70% of network traffic.

- **Information-centric and Intent-Based Networks (ICNs):**

Internet Research Task Force (IRTF) and Internet Engineering Task Force (IETF) have begun their research in ICNs. ICNs separate the content from the location identifier as it uses abstract naming convention rather than IP addressing. Even 5G-based proposals were put forth for ICNs. By 2030, ICNs will be the driving force for networks. Also, intent-based networking and service design provide a plausible solution for operational management and lifecycle management for the network infrastructure that is the center of 6G. Real-time optimization, continuous monitoring, and adaptation to load fluctuation and service state are the prime objectives of future networks.

- **360-Cybersecurity and Privacy-By-Engineering Design:**

6G networks call for a 360-Cybersecurity. The present protocols and access control methods are inefficient and susceptible to attacks. The need for a robust standardization of end-to-end solutions based on top-down approaches mandates a novel network setting. The need for a Privacy-by-Engineering design will imbibe privacy protection within the protocol architecture, i.e., privacy-based package forwarding. The 6G paradigm will ensure contextual level security.

- **Future-Proofing Emerging Technologies:**

Technology is ever-evolving. With the new swamp of technology on the horizon, we require a network that can efficiently embed these and satisfy their requirements. Also, some of these technologies will make 6G communication more efficacious, as quantum can make communication tamper-proof. DLTs will make the transfer immutable. Explainable AI (xAI) will aid in regulatory requirements rather than solve consumer-related issues.

Organization of Paper The paper is accentuated as follows: We begin by discussing the architecture of 6G that explains to the readers the merits of 6G over 5G. Then in the next section, we dive into the IoT-based application that is the incentive for 6G development. With this, we move to the challenges faced by 6G and how AI can help to overcome them, citing some recent methods proposed for the same. Then we head towards the discussion of problems faced in data collection. With this, we proceed to the ML-integrated 6G network. Moreover finally, we discuss the application domain of 6G. We conclude the paper by citing the case study of Hexa-X undertaken as a European joint venture.

2 Literature Review

The novice network paradigm of 6G coupled with IoT and AI has been a focus of researchers. Huang et al. [10] provided an evolutionary study of networks and analyzed the challenges in ubiquitous coverage promised by 6G and the potential technologies that might provide breakthrough solutions. Akhtar et al. [11] emphasized quantum communication, machine learning, blockchain, and shared spectrum technologies as potential candidates for intelligent network generation. The prime focus of their work was e-health care, bio-sensing, HC, and IoT applications powered by 6G. Tataria et al. [12] discussed the 6G-enabled HC, tactile Internet, edge computing, and IoT. The analysis does not consider the AI and IoT efficient 6G network coverage. Giordani et al. [13] surveyed the potential requirements and system-level perspective on 6G. They presented several key performance indicators for evaluating the efficiency of 6G. Chowdhury et al. [14] presented a vision of 6G wireless communication and the application insight into emerging technologies such as cell-free communication, backhaul networks, dynamic network slicing, proactive caching, etc. They worked on the plausible advantage of these technologies powered by 6G.

Some studies based on AI that use a down-top approach for 6G architecture have also been proposed in the past few years. Yang et al. [15] proffered a down-top approach of 6G architecture in combination with AI. Their prime objective was radio network resource management. Alex Mathew [16] proposed an AI-empowered architecture engineering for 6G for specific application needs in mechanic system adjustment and smart source management. These application areas are quite complicated as the network here is segmented into four divisions: smart application layer, intelligent control layer, information search, and logic layer, and intelligent sensing layer. The combination of AI could aid in smart spectrum management, handover management, intelligent mobility, and edge computing. These findings could provide future research directions in hardware developments and energy management. Letaief et al. [17] worked on design and optimization technology for smart 6G networks. A similar attempt was made in [18]. Zhang and Zu surveyed AI-enabled 6G networks with an emphasis on intelligent traffic control, resource management, and energy optimization. Their work neglected the requisites of AI-based applications and hence could not foresee the impediments caused. Kato et al. [19] analyzed the importance of Artificial Intelligence in network intelligentization. The issue of automatically configured cellular communication systems, machine learning architectures, and computation efficiency is also addressed. Ismail et al. [20] provided a holistic analysis of self-learning models powered by AI and 6G. They also investigated the impact of IoT-driven methods in the 6G communication paradigm. The work discussed the chronological evolution of the communication system that paved the way for 6G. Their main focus was smart city development and smart ecosystem management.

Security in the 6G network is another intriguing field. The ubiquitous network coverage encompasses heterogeneous resources. Manogaran et al. [21] proffered

Table 2 Comparison of literature works studied

Year	Author	Merit	Demerit
2020	Akhtar et al. [11]	Evolution studies, IoT coupled 6G	AI-enabled 6G is not considered
2021	Tataria et al. [12]	Top-down approach over 6G architecture	Self-learning models powered by 6G are untouched
2020	Yang et al. [15]	Down-top approach for 6G architecture discusses the AI-enabled networks	IoT and technology-empowered 6G applications are not considered
2019	Letaief et al. [17]	Down-top approach	HC and tactile network possibility not illustrated
2020	Zhang and Zu [18]	AI-based 6G considered	Evolution studies not included, and technology basis weak
2022	Ismail et al. [20]	The AI-enabled study, self-learning models considered	

blockchain-based integrated security measures (BISM) for security and privacy preservation over a 6G network. The Q-learning procedure was used to decide on access delegation and denial-related decisions. Hakeem et al. [22] investigated the security requirements and trust issues in 6G networks. Further, they proposed a 6G security architecture and the required improvements in the present 5G-based security infrastructure. The work furnishes security evolution over time in legacy mobile networks and also the challenges that might crop up in 6G. The various present solutions are evaluated against the trustworthiness and access control for reliable and rapid networks such as 6G. Table 2 summarizes the efficiencies and limitations of the various libraries that have been studied.

3 The Architecture of 6G

We anticipate that 6G will bring a paradigm shift in wireless networks, especially in coverage, spectrum, applications, and security. As in [23], 6G will provide a “3D” communication network that will augment terrestrial communication along with NTN’s [24] that will cover the sea and space as well. With this coverage expansion, communication is expected to grow across the terrestrial space boundaries. Figure 2 depicts the envisioned 6G network paradigm.

Key Performance Indicators As described earlier, Key Performance Indicators (KPIs) for 6G were vastly different from the metrics for 5G. The metrics are categorized into four categories: KPIs for evolution capabilities, new e2e measures, New capability areas, and Key value indicators. Figure 3 depicts the performance parameters for 6G.

Fig. 2 6G envisioned paradigm shift

Coverage	Spectrum	Application	Security
<ul style="list-style-type: none"> •Terrestrial •UAV •Space •Maritime 	<ul style="list-style-type: none"> •Sub 6-Ghz •mmWave •terahertz •optical bands 	<ul style="list-style-type: none"> •IoT to IoS based •Critical IoT based •braodband based •Industrial IoT based 	<ul style="list-style-type: none"> •Data driven •AI based •ML based •Distributed learning

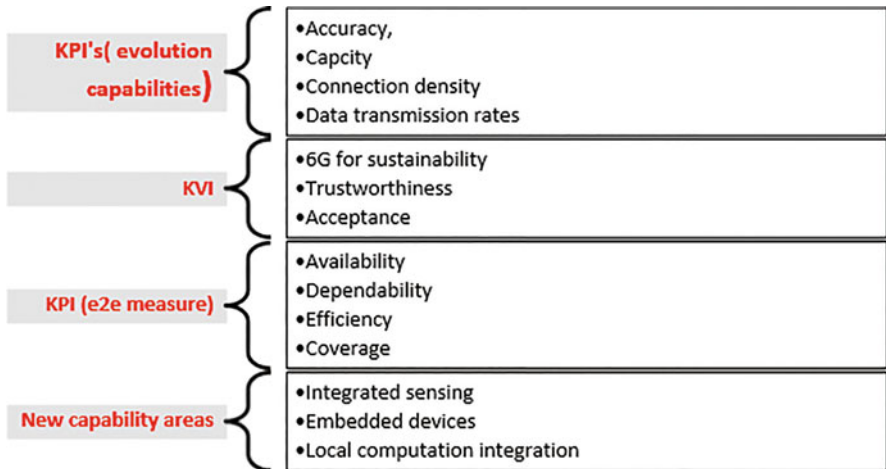


Fig. 3 KPIs for 6G

Figure 4 illustrates the perceptive architecture of the 6G network. In this architecture, we consider Machine Learning, Artificial Intelligence, the Internet of Things, and the Internet as the four pillars of the communication network. The whole paradigm provides the communication network in three major dimensions: terrestrial, maritime, and space. The network can be used on the terrestrial surface for various purposes like smart cities, smart homes, healthcare, etc. Each of these domains is explained in the sections to come. Further, the maritime communication supported by the 6G network ensures unimpeded data transmission for sea and oceans. 5G network exhibits limited facility in this context. The third communication facility is communication through space. The quest to explore celestial bodies has gained new momentum with the growing technology. 6G, with its proficient data transmission services, will also accelerate man’s reach in space. The architecture of the 6G communication paradigm shows its ability to cut across boundaries and provide ubiquitous coverage.

Elements of 6G Networks The core elements of 6G networks are shown in Fig. 5. Out of all the elements, the air interface is of prime importance. Orthogonal fre-

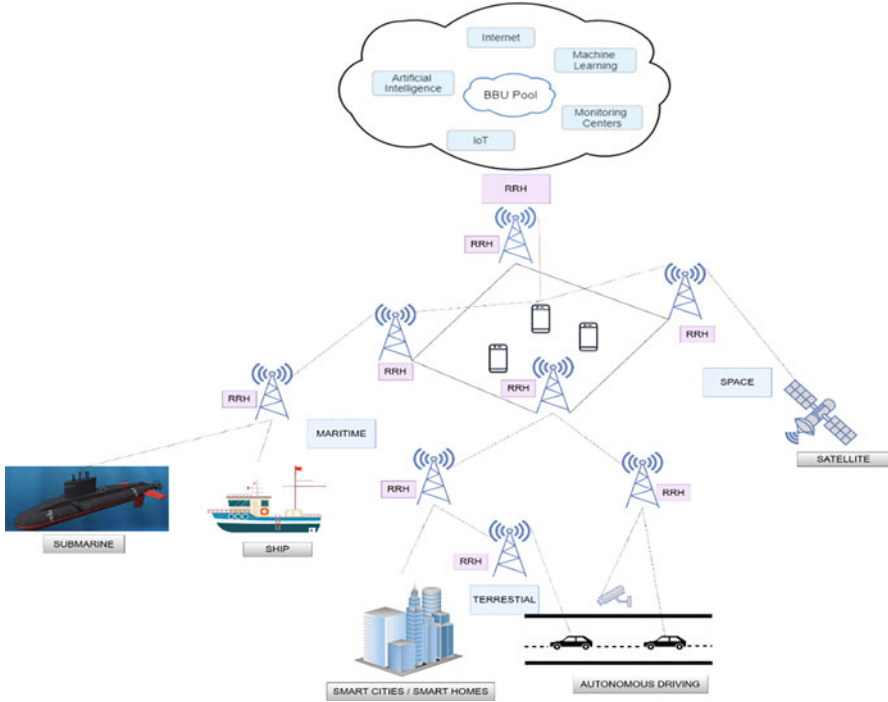


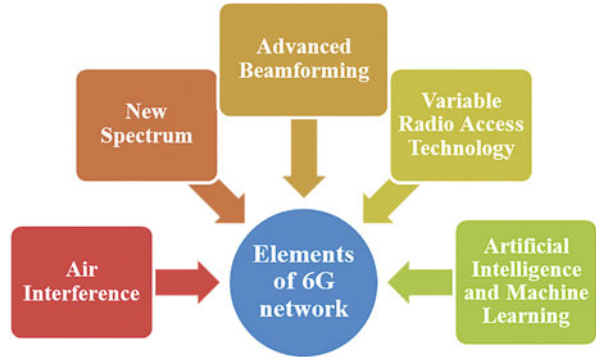
Fig. 4 6G architecture framework

quency division multiplexing (OFDM) and Code division multiple access (CDMA) was the kingpin in 4G and 3G, respectively. Thus, for 6G, a new interface would be the need of the hour. Researchers have proposed non-orthogonal multiple access (NOMA) and rate-splitting multiple access (RSMA) as plausible candidates for air interfaces supporting 6G. An AI-based intelligent air interface would suffice for the medium intelligentization. The 3GPP current release can also support the 6G with modifications in symbol durations, sub-carrier spacing, self-configuration, channel conditions, service requirements, and phase noise.

The second element of the 6G network includes the proposal of a new spectrum. mmWave is a potential candidate for 5G, but it requires several improvements to merge personal BSs and satellite connectivity in cellular communication. Thus the possibility of an unlicensed spectrum like the use of mmWave, THz, and visible light spectrum simultaneously in combination with Multiple Inputs and Multiple Outputs (MIMO) is being investigated.

The next important element is an advanced beamforming-based large-scale antenna(VLSA). The objective is the uni-directional diversion of energy toward the user. Intelligent reflecting surfaces (IRs) can reflect electromagnetic waves by phase adjustment. They provide low-power and low-cost passive elements with reduced hardware complexity at the transmitter and receiver sites. This makes it a

Fig. 5 6G network elements



potential beamforming candidate for 6G networks. The orbital angular momentum property of electromagnetic waves allows an unlimited number of orthogonal modes that facilitate multiple data streams over channels. Also, they can be modeled for narrowband and wideband frequency hopping schemes, and this property facilitates efficient beamforming capability.

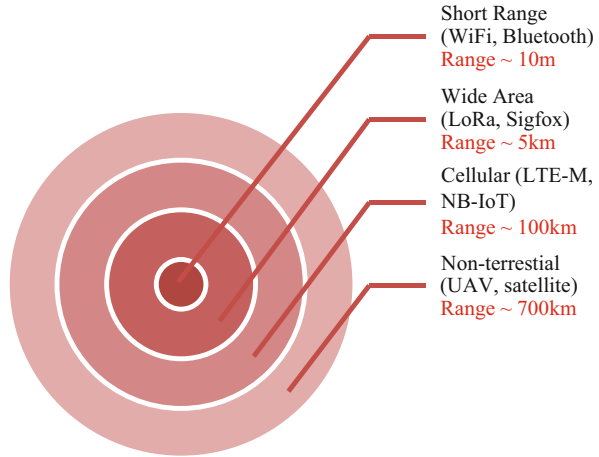
The coexistence of variable radio access technologies facilitates 6G with wide infrastructure support that provides ubiquitous support, high throughput, low latency, and massive connectivity. The adaptability to new services like quantum communications, AI/ML, quantum machine learning (QML), etc., along with the integration of intelligent cloudification, software nation, slicing, and virtualization, is anticipated to be an important feature of 6G. The various KPIs for THz communication, orbital angular momentum multiplexing, spatial Modulation, and intelligent re-configurable reflecting surfaces are also suggested for the 6G-based communication paradigm shift.

The self-learning models are the foundation of digital transitions. The advent of 5G has complicated the scenario, and the anticipation of 6G has instigated new challenges. Incorporating AI-driven channel intelligentization will score on KPIs. Further, the issues of resource management, lifecycle management, anomaly detection, fault detection, etc., can be dealt with effectively with the inclusion of machine learning and deep learning-based algorithms.

4 IoT and 6G

IoT data is characterized as large-scale streaming data classified according to 6Vs, volume, velocity, variability, veracity, variety, and value. The wireless network is critical for IoT technologies. IoT network is classified in terms of range as per Fig. 6. With the rapid advancements in IoT devices, many applications like holographic communications, five-sense communications, wireless brain-computer interfaces

Fig. 6 IoT network classification



(WBCI), etc., have gained momentum. The amalgamation of 6G and IoT greatly incentivizes these applications.

Holographic Communication Holographic communication involves capturing images of people or objects in the real-time present at some location and transmitting them to another location so that the projected image gives the illusion of that object/person being present right in front of the user [25]. uRLLC, eMMB, and mMTC triad of 5G network fail to provide low latency transmission of a large amount of data in holographic communication.

Five-Sense Communication A similar issue is witnessed in five-sense communications [26], where the five-sense media captures the information about the five senses of humans and transfers them to a remote location. These multi-sensory applications and VR/AR require immersive transmission services. With its terrestrial and space localization interface, 6G best serves the purpose.

WBCI or WMMI Another potent application of IoT in combination with 6G is the wireless brain-computer interface WBCI/Wireless mind-machine interfaces WMMI. In this application, human thoughts play a kingpin role in communication with machines [27]. Here, the human neurotic signals are read using electrodes and then translated into machine-understandable code [28]. WBCI is the future of communication heuristics as it uses acquired human brain signals to communicate with external devices and perform daily operations on machines. This approach finds its use in healthcare, homes, smart city development, multi-brain-controlled cinema, etc. [29]. This IoT application includes tactile Internet, haptics, and neurological communications and relies heavily on the 6G networks as ultra-low latency, data sensitivity, quality services, and the quantity of experience (QoE) are some of the prima-facie requirements of WBCI.

Smart Education/Training 6G is also imperative from the perspective of smart education and training purposes. Coupling 6G with the abovementioned applications facilitates the students to visualize 3D models, access eminent teachers from remote locations, and learn new skills remotely. Also, it will help design more interactive classes and simpler ways of taking exams using sensors to collect the data. Besides educating, it is also vital for training purposes. Holographic communication can help to train students in remote places without being exposed to a dangerous environment.

Industry and 6G The advent of the Internet has revolutionized industrial output. 6G, with its enhanced capability coupled with AI, would aid in the complete automation of process control and decision-making. While intelligent decision-making is AI's onus, error-free data transfer is a 6G network task. Also, with holographic communication, WBCI, one can easily undertake maintenance work remotely and solve problems without being physically present at the site. This use case requires low latency, broadband, and reliable communication and transmission facility, as anticipated in 6G.

Autonomous Driving Sustainable development has been approved as an agenda to be accomplished by 2050. With this, fully autonomous driving comes to the forefront. Employing multiple high-definition cameras and a high-precision sensor coupled with 6G, it aims to annihilate the need for a vehicular cockpit; hence, the vehicles will perform all the driving tasks autonomously. This would be a breakthrough for differently abled people. This system has a three-layered structure: perception, planning, and control [30]. The information transferred via sensors like image sensors, multi-meter wave radar, and Light Detection and Ranging (LIDAR) is fed to the perception layer. At the same time, the planning layer involves conditional assessment for stopping, accelerating, and overtaking. The decision-making process brings in the control layer functionalities like gear, brake, and throttle. Designing process must cater to two things. First, the real and laboratory worlds must not be too far from each other. Second, there exist different driving situations in different places. To allow autonomous driving, there must be safety measures to deal with them. Hence, this is the epitome of AI, MEC, and 6G coupled IoT data processing. Many researchers have also proposed the inclusion of blockchain in this scenario.

Smart Cities A smart city is one where the process of decision-making, planning, and administration is intelligently run autonomously by procuring and processing volumes of data from heterogeneous sources like urban planning, garbage collection, traffic prediction, location prediction, etc., it leverages the burden of data collection and maintenance. 6G, with its fast data collection and low latency transmission, will support the smart city vision. Further, a smart home equipped with all internet-connected devices and appliances relies on the decision-making attributes of intelligent systems. Like WBCI, people can control household appliances with voice or brain signals. This is another combined onus of 6G and AI-based over IoT

devices. We must countermeasure the issues of connectivity and coverage over the network to realize the dream of smart cities and smart homes.

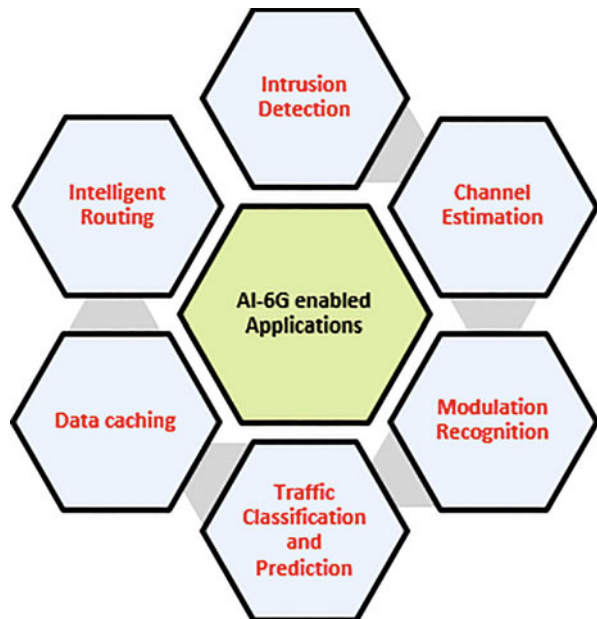
5 6G Challenges and AI Implications

With the shift in communication paradigm, we also anticipate new applications at the horizons that require fast, reliable, and intelligent future-generation communication networks. 6G coupled with Artificial intelligence can easily exploit this utility [31–33]. Unlike its predecessors, 6G can be used in combination with AI to accommodate varied spectra of applications like uMUB, uHSLLC, mMTC, and uHDD. AI will supplement data efficiency, reduce delays and potential security risks, and support faster communication network computation [34].

Several applications IoV, IoMTs, IoD, IoRT, IIoT, etc., [35–37] have put the rigorous quality of services requirements, and their AI-Big data-driven features make 5G network highly unsuitable for them. 6G must provide facilities beyond 5G LTE, combining context-aware algorithms to make the system more efficacious. Figure 7 depicts the various AI-6G-enabled applications that will be the focus of this section.

Channel Estimation With the requirement of high data rates (Tbps), low latency (order 0.1–1 ms), 6G radio access requires Terahertz communication [38], ultra-massive multiple-input multiple-output (MIMO) [39] and Intelligent surfaces [40],

Fig. 7 AI-6G-enabled applications



all of which are yet in their infancy stage. These developments will put extra pressure on communication channels; hence, the channel estimation for appropriate communication is a complex task. Channel estimation inhibits the channel effect by estimating the channel characteristics appropriate for recovering the transmitted information. Ye et al. [41] posited Deep Neural Network based channel estimation system in OFDM. Ago et al. [42] proposed CNN based approach over MIMO.

Modulation Recognition Modulation recognition is the process to identify the modulation information under the influence of noise for better data transmission [43]. For a 6G network defining ideal modulation information is complex. It requires a comprehensive knowledge of the challenges and analytical clarity. For 5G and its predecessor, decision theory-based and statistical pattern recognition-based approaches have succeeded. However, similar advancements for 6G are futile owing to high transmission rates and broad spectra.

Further, it also requires appropriate estimation of user payload. To overcome such challenges, intelligent 6G is a must. Zhang et al. [44] put forward CNN and LSTM-based models for the same. Yang et al. proposed the CNN and RNN-based approach. In contrast, Shi et al. [45] proposed CNN-based federated learning for modulation recognition and privacy preservation.

Traffic Classification and Prediction As the number of IoT users increases, the data generated increases. A large amount of traffic should be categorized under different classes to ensure quality services, resource management, and security. There are two categories of methods for the same: payload-based method and port-based method [46]. Ren et al. [47] proposed a Tree-RNN-based method for traffic classification into 12 classes. Martin et al. [48] proposed hybrid RNN and CNN-based approaches for IoT devices. With the 6G network in foresight, we require adequate methods to predict future traffic, as this will help manage the network without much stalling and resource disputes. Vinay Kumar et al. [49] compared the performance of different RNNs for traffic prediction. Aloraifan et al. [50] proposed Bidirectional LSTM (Bi-LSTM) and Bidirectional -GRU (Bi-GRU) based approaches for the same.

Data Caching With the growing data generation, data storage is also a big challenge. For the 6G spectrum, the data transmission needs to devise a method for data caching and avoid overloading intelligently. Edge caching [51] based approach was appropriate for a static network but insufficient for a dynamic one. Jiang et al. [52] proposed a deep Q-learning-based and Zhang et al. [53] put forward the DRL-based approach, both of which were the enhanced versions of edge caching.

Intrusion Detection 6G network communication is expected to be rapid, confidential, and reliable. Assuring security is challenging in an IoT environment as numerous sensors are employed. Thus, AI-based approaches need to be incorporated to assure security. Sharifi et al. [54] proposed K Nearest Neighbours (KNN) and K means-based approaches for intrusion detection. Yin et al. [55] deployed RNN for multi-class intrusion detection.

Intelligent Routing Routing Techniques are quintessential to ensuring QoS services of 6G. Traditional protocols deploying meta-heuristics approaches are relatively inefficient with the 6G paradigm. Tang et al. [56] proffered real-time deep CNN-based routing protocol for a mesh topology. Liu et al. [57] put forward DRL based approach for the same, which used CNN for routing purposes.

6 Challenges in Data Collection

Data Security IoT devices collect highly sensitive information about the users. As internet-connected cameras, voice assistants, and similar tools can monitor users' activities and conversations. The IoT devices access the public Internet to send the data to cloud servers for processing. As a result, they have poor security. IoT security issues:

- **Weak Password:**

IoT devices are deployed with default, weak, and hardcoded passwords. As a result, Malicious hackers can exploit password security and gain access to the devices.

- **Unpatched Flaws:**

IoT vendors are uncontrolled and use inefficient secure development processes, which results in vulnerable products being shipped.

Data Privacy IoT devices process and gather many data, covering various data privacy rules. IoT device manufacturers and users must protect it per applicable laws.

Data Volume IoT devices generate enormous volumes of data. IoT devices produced approximately 18.3 zettabytes of information in 2019, which is predicted to increase to 73.1 zettabytes by 2025. IoT devices are installed in remote regions with poor Internet connectivity, which makes it challenging and expensive to transfer the data gathered. The cloud servers must quickly process and analyze data quantities to draw crucial conclusions and communicate to the IoT devices.

Data Complexity IoT devices collect as much information as possible and send it to cloud-based servers for processing, which creates complex datasets. The data produced by IoT devices is unstructured and provides a limited perspective [58].

7 Importance of Machine Learning in 6G

Integrating ML techniques into the 6G wireless network communication paradigm will make the technology more economical and optimized. ML-based approaches include classification and regression to integrate the 6G with the intelligent system

architecture. ML focuses on pattern recognition in data and eventually uses it for classification. The radio spectrum demand is accelerating, as is the data traffic. To successfully cope with such issues, deploying the mix of ML-based 6G over IoT devices is necessary. Liyanage et al. [59] analyzed the applicability of ML to cope with zero-touch operations in real-time events. This will facilitate greater control and better resource management on 6G networks. The contribution of ML to 6G is categorized at different levels. At the physical layer level, we aim to use ML to solve issues that cannot be solved so far due to non-linearity issues, like uplink and downlink reciprocity in FDD, interference detection and mitigation, channel protection, etc. One of the primary reasons for performance degradation of the physical layer of the communication model is the occurrence of non-linear factors with the 5G networks. Hence ML methods will prove to be a better solution for the 6G paradigm. ML methods can alleviate the need to design decoding, modulation, and waveform separately. Instead, a simple end-to-end mapping can serve the purpose. For successful use of an IoT device, it must have time/frequency and cell synchronized to cater to the requirements of 4G and its successors. For carrier frequency offset (CFO), an end-to-end-encoder (AE) based system on implementing a deep neural network for sync signal, as proposed in [60], serves the purpose.

Further, to solve the issue of breaks in the presence (BIPs) in VR image transmission, the Federated Echo State Network (ESN) algorithm is used to ascertain the orientation and mobility of the user [61]. Also, the ML solutions can be used for uplink grants in communication with IoT devices at low mobility [62]. Such an arrangement is called machine-type communications (MTC).

8 Applications of 6G in Different Domains

Every emerging technology imposes a new dimension of applications. In this section, we outline potential 6G technology applications.

Multi-sensory Applications VR/AR technology has been extended to the 5G network. The limited capability of 5G has stunted the growth of VR/AR. 6G will be able to address those issues with eMMB and uRLLC.

Connected Robotics and Autonomous System The field of robotics requires high network support for full autonomous functioning. With limited 5G capabilities, this is unachievable. Hence, 6G is best suited for such services as it supports a multidimensional network with the capability to integrate AI/ML within the network. Strianti et al. [26] preferred network resource control, caching, and automatic handling. In their attempt to address the issue of limited autonomy, they used cloud services, databases, and UAVs to achieve complete autonomy. 6G has the immense capability for maritime robotics as well.

Internet of Nano-things The field of nanotechnology is fast growing. It has led to the development of nanosensors. IoNT integrates nanotechnology with IoT [63], and

ever since its development, it has been deployed in combination with big data, cloud computing, etc. With the limited storage and memory, IoNT faces many challenges, that 6G can resolve.

Intelligent Internet of Medical Things (IIoMT) The future of medical science is bright with the possibility of telesurgery across boundaries with the data-intensive and delay-sensitive characteristics of 6G. As mentioned in the previous sections holographic communication and VR/AR technology have a large scope of development under 6G. They can also be very useful in surgical cases. Secure and protected storage of the medical data of patients and even easy and fast transmission of the same using 6G would ensure better health services.

9 Case Study of 6G Network

Hexa-X- the joint European initiative to shape 6G has started envisaging the 6G as the potential substitute for 5G and its technological aspects along with the research voids that might exist. A few of the key developments of Hexa-X are discussed in this section.

6G Use Cases Hexa-X has categorized the 6G use cases into five different heads.

- **Sustainable development:**

6G is envisioned to provide massive opportunities to achieve sustainable development goals. For this, the responsibility of data procurement and storage is to be done globally. 6G combined with AI and ML will provide for the global outreach of these goals.

- **Local trust zones:**

Fields such as E-healthcare, and smart cities development are new and merging use cases of 6G. they require extensive use of IoT micro-networks for collecting massive data and keeping the confidentiality of the data intact. IoT devices can automatically connect in a mesh topology, but they rely on the local leased spectrum. 6G here can facilitate communication and annihilate the need for custom-built networks.

- **Robots to cobots:**

Cooperative, mobile robots (“cobots”) are the new trend immersive due to the advancement in AI technology. 6G is anticipated to provide a communication paradigm coupled with AI and ML to facilitate the software requirements and interaction intricacies.

- **Massive twinning:**

The digital twinning concept is currently deployed over a 5G network but with limited operationality. 6G, with low latency and high reliability in data transmission,

is anticipated to facilitate massive twinning at a large scale. The information can be used for real-time health, needs, crops, and livestock monitoring.

- **Telepresence:**

By 2030, Hexa-X anticipates that telepresence will be far more in reach and that communication with digital replicas will be easy, even remotely.

Radio Performance Hexa-X analyses the sub-THz frequencies waveform and modulation, channel characterization, and hardware feasibility from a 6G perspective. It also studies mm-wave usage for low-cost and energy-efficient solutions with <1 cm precision. We can visit the radio performance aspects of 6G from Hexa-X's recent release, "[Towards Tbps communications in 6G: Use cases and gap analysis](#)". The key technologies for Tbps/THz radio communication as per Hexa-X are.

- Mobility synchronization
- Coverage capacity
- Flexibility hardware complexity
- Energy efficiency/Spectrum efficiency
- Signal quality range.

Smart and Intelligent Orchestration for 6G Network AI/ML technology coupled with a 6G network supports high accuracy, less energy consumption, and highly reliable sources. AI can be considered the "brain" of the network. It allows the network to adapt to the dynamic traffic demands and predictive orchestration mechanism. Hexa-X released a critical analysis of network orchestration entitled "Gaps, features, and enablers for B5G/6G service management and orchestration".

Integration Dynamics In real-time, 6G is expected to integrate heterogeneous devices for data collection at the global scale. A prior study of all the specialized domains is a must to accommodate such a vast arrangement. Hexa-X published an initial exploration of requirements entitled "Gap analysis and technical work plan for special-purpose functionality."

10 Conclusion and Future Scope

With 5G in its infancy, 6G has been envisioned to support high data-intensive applications with the least data latency. Many emerging technologies have also posed direct demand for fast and reliable communication networks over the terrestrial, maritime, and space. In this paper, we have analyzed the limitations of 5G that have led to the anticipation of 6G and how the data procured from IoT devices can be used over intelligent 6G, which is the combination of the 6G network and Artificial Intelligence. AI/ML-enabled 6G networks can address several upcoming

communication network issues and substantiate the decision-making process. The QoS and SLA requirements of the network can also be satiated by intelligent 6G. Then we explore the different domains of applications for 6G and the probable use cases with the futuristic perception. In the end, we discussed the case study of Hexa-X, a European joint venture. Thus, in this paper, we have clarified the vision of 6G, the probable use cases, and the methods to overcome the issues that might impede the future communication paradigm shift.

References

1. Kurt, G. K., Khoshkholgh, M. G., Alfattani, S., Ibrahim, A., Darwish, T. S., Alam, M. S., et al. (2021). A vision and framework for the high altitude platform station (HAPS) networks of the future. *IEEE Communications Surveys & Tutorials*, 23(2), 729–779.
2. Hossain, M. S., & Muhammad, G. (2017). Emotion-aware connected healthcare big data towards 5G. *IEEE Internet of Things Journal*, 5(4), 2399–2406.
3. “5G: The future of IoT”. (2019). <https://www.5gamericas.org/wp-content/uploads/2019/07/5G-Americas-White-Paper-on-5G-IOT-FINAL-7.16.pdf>
4. Koohang, A., Sargent, C. S., Nord, J. H., & Paliszkiwicz, J. (2022). Internet of Things (IoT): From awareness to continued use. *International Journal of Information Management*, 62, 102442.
5. Kamruzzaman, M. M., Alrashdi, I., & Alqazzaz, A. (2022). New opportunities, challenges, and applications of edge-AI for connected healthcare in Internet of medical things for smart cities. *Journal of Healthcare Engineering*, 2022, 14. Article ID 2950699. <https://doi.org/10.1155/2022/2950699>
6. Qi, Q., Chen, X., Zhong, C., & Zhang, Z. (2020). Integration of energy, computation and communication in 6G cellular internet of things. *IEEE Communications Letters*, 24(6), 1333–1337.
7. Rout, D., Mishra, S. J., Ota, R., & Gupta, P. (2021). Customer satisfaction towards internet speed of various telecom service providers: An exploratory study in Bhubaneswar. *International Journal of All Research Education and Scientific Methods (IJARESM)*, 9(3), 1463–1473.
8. Ahmad, I., Shahabuddin, S., Kumar, T., Okwuibe, J., Gurto, A., & Ylianttila, M. (2019). Security for 5G and beyond. *IEEE Communications Surveys & Tutorials*, 21(4), 3682–3722.
9. Yang, R., Yu, F. R., Si, P., Yang, Z., & Zhang, Y. (2019). Integrated blockchain and edge computing systems: A survey, some research issues and challenges. *IEEE Communications Surveys & Tutorials*, 21(2), 1508–1532.
10. Huang, T., Yang, W., Wu, J., Ma, J., Zhang, X., & Zhang, D. (2019). A survey on green 6G network: Architecture and technologies. *IEEE Access*, 7, 175758–175768.
11. Akhtar, M. W., Hassan, S. A., Ghaffar, R., Jung, H., Garg, S., & Hossain, M. S. (2020). The shift to 6G communications: Vision and requirements. *Human-centric Computing and Information Sciences*, 10(1), 1–27.
12. Tataria, H., Shafi, M., Molisch, A. F., Dohler, M., Sjöland, H., & Tufvesson, F. (2021). 6G wireless systems: Vision, requirements, challenges, insights, and opportunities. *Proceedings of the IEEE*, 109(7), 1166–1199.
13. Giordani, M., Polese, M., Mezzavilla, M., Rangan, S., & Zorzi, M. (2020). Toward 6G networks: Use cases and technologies. *IEEE Communications Magazine*, 58(3), 55–61.
14. Chowdhury, M. Z., Shahjalal, M., Ahmed, S., & Jang, Y. M. (2020). 6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions. *IEEE Open Journal of the Communications Society*, 1, 957–975.

15. Yang, H., Alphones, A., Xiong, Z., Niyato, D., Zhao, J., & Wu, K. (2020). Artificial-intelligence-enabled intelligent 6G networks. *IEEE Network*, 34(6), 272–280.
16. Mathew, A. (2021). Artificial intelligence and cognitive computing for 6G communications & networks. *International Journal of Computer Science and Mobile Computing*, 10(3), 26–31.
17. Letaief, K. B., Chen, W., Shi, Y., Zhang, J., & Zhang, Y. J. A. (2019). The roadmap to 6G: AI empowered wireless networks. *IEEE Communications Magazine*, 57(8), 84–90.
18. Zhang, S., & Zhu, D. (2020). Towards artificial intelligence enabled 6G: State of the art, challenges, and opportunities. *Computer Networks*, 183, 107556.
19. Kato, N., Mao, B., Tang, F., Kawamoto, Y., & Liu, J. (2020). Ten challenges in advancing machine learning technologies toward 6G. *IEEE Wireless Communications*, 27(3), 96–103. [9061001]. <https://doi.org/10.1109/MWC.001.1900476>
20. Ismail, L., & Buyya, R. (2022). Artificial intelligence applications and self-learning 6G networks for smart cities digital ecosystems: Taxonomy, challenges, and future directions. *Sensors*, 22(15), 5750.
21. Manogaran, G., Rawal, B. S., Saravanan, V., Kumar, P. M., Martínez, O. S., Crespo, R. G., et al. (2020). Blockchain based integrated security measure for reliable service delegation in 6G communication environment. *Computer Communications*, 161, 248–256.
22. Abdel Hakeem, S. A., Hussein, H. H., & Kim, H. (2022). Security requirements and challenges of 6G technologies and applications. *Sensors*, 22(5), 1969.
23. You, X., Wang, C. X., Huang, J., Gao, X., Zhang, Z., Wang, M., et al. (2021). Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts. *SCIENCE CHINA Information Sciences*, 64(1), 1–74.
24. Shin, W., & Vaezi, M. (2021). UAV-enabled cellular networks. In *5G and beyond* (pp. 165–200). Springer.
25. Huang, C., Hu, S., Alexandropoulos, G. C., Zappone, A., Yuen, C., Zhang, R., et al. (2020). Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends. *IEEE Wireless Communications*, 27(5), 118–125.
26. Strinati, E. C., Barbarossa, S., Gonzalez-Jimenez, J. L., Ktenas, D., Cassiau, N., Maret, L., & Dehos, C. (2019). 6G: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication. *IEEE Vehicular Technology Magazine*, 14(3), 42–50.
27. Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113, 767–791.
28. Guruprakash, S., Balaganesh, R., Divakar, M., Aravinth, K., & Kavitha, S. (2016). Brain controlled home automation. *International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)*, 2(10), 430–436.
29. Saad, W., Bennis, M., & Chen, M. (2019). A vision of 6G wireless systems: Applications, trends, technologies, and open research problems. *IEEE Network*, 34(3), 134–142.
30. Pendleton, S. D., Andersen, H., Du, X., Shen, X., Meghjani, M., Eng, Y. H., et al. (2017). Perception, planning, control, and coordination for autonomous vehicles. *Machines*, 5(1), 6.
31. Jagannath, A., Jagannath, J., & Melodia, T. (2021). Redefining wireless communication for 6G: Signal processing meets deep learning with deep unfolding. *IEEE Transactions on Artificial Intelligence*, 2(6), 528–536.
32. Stoica, R. A., & de Abreu, G. T. F. (2019). *6G: The wireless communications network for collaborative and AI applications*. arXiv preprint arXiv:1904.03413.
33. Zhao, J. (2019). *A survey of intelligent reflecting surfaces (IRSs): Towards 6G wireless communication networks*. arXiv preprint arXiv:1907.04789.
34. Mahmood, N. H., Alves, H., López, O. A., Shehab, M., Osorio, D. P. M., & Latva-Aho, M. (2020, March). Six key features of machine type communication in 6G. In *2020 2nd 6G wireless SUMMIT (6G SUMMIT)* (pp. 1–5). IEEE.
35. Ismail, L., Hagimont, D., & Mossiere, J. (2000). *Evaluation of the mobile agents technology: Comparison with the client/server paradigm* (p. 19). Information Science and Technology (IST).

36. Hagimont, D., & Ismail, L. (2000). Agents mobiles et client/serveur: évaluation de performance et comparaison. *Technique et science informatiques*, 19(9), 1223–1244.
37. Ismail, L., & Belkhouche, B. (2009, June). Full and autonomic mobility management for Mobile agents. In *2009 first international conference on advances in future Internet* (pp. 31–38). IEEE.
38. Akyildiz, I. F., Jornet, J. M., & Han, C. (2014). Terahertz band: Next frontier for wireless communications. *Physical Communication*, 12, 16–32.
39. Sareddeen, H., Alouini, M. S., & Al-Naffouri, T. Y. (2019). Terahertz-band ultra-massive spatial modulation MIMO. *IEEE Journal on Selected Areas in Communications*, 37(9), 2040–2052.
40. Basar, E. (2019, June). Transmission through large intelligent surfaces: A new frontier in wireless communications. In *2019 European Conference on Networks and Communications (EuCNC)* (pp. 112–117). IEEE.
41. Ye, H., Li, G. Y., & Juang, B. H. (2017). Power of deep learning for channel estimation and signal detection in OFDM systems. *IEEE Wireless Communications Letters*, 7(1), 114–117.
42. Gao, J., Hu, M., Zhong, C., Li, G. Y., & Zhang, Z. (2021). An attention-aided deep learning framework for massive MIMO channel estimation. *IEEE Transactions on Wireless Communications*, 21(3), 1823–1835.
43. Zhang, M., Zeng, Y., Han, Z., & Gong, Y. (2018, June). Automatic modulation recognition using deep learning architectures. In *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)* (pp. 1–5). IEEE.
44. Yang, C., He, Z., Peng, Y., Wang, Y., & Yang, J. (2019). Deep learning aided method for automatic modulation recognition. *IEEE Access*, 7, 109063–109068.
45. Shi, J., Qi, L., Li, K., & Lin, Y. (2021). Signal modulation recognition method based on differential privacy federated learning. *Wireless Communications and Mobile Computing*, 2021, 1–13.
46. Finsterbusch, M., Richter, C., Rocha, E., Muller, J. A., & Hanssgen, K. (2013). A survey of payload-based traffic classification approaches. *IEEE Communications Surveys & Tutorials*, 16(2), 1135–1156.
47. Ren, X., Gu, H., & Wei, W. (2021). Tree-RNN: Tree structural recurrent neural network for network traffic classification. *Expert Systems with Applications*, 167, 114363.
48. Lopez-Martin, M., Carro, B., Sanchez-Esguevillas, A., & Lloret, J. (2017). Network traffic classifier with convolutional and recurrent neural networks for Internet of Things. *IEEE Access*, 5, 18042–18050.
49. Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2017, September). Applying deep learning approaches for network traffic prediction. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 2353–2358). IEEE.
50. Aloraifan, D., Ahmad, I., & Alrashed, E. (2021). Deep learning based network traffic matrix prediction. *International Journal of Intelligent Networks*, 2, 46–56.
51. Liu, D., Chen, B., Yang, C., & Molisch, A. F. (2016). Caching at the wireless edge: Design aspects, challenges, and future directions. *IEEE Communications Magazine*, 54(9), 22–28.
52. Jiang, F., Yuan, Z., Sun, C., & Wang, J. (2019). Deep Q-learning-based content caching with update strategy for fog radio access networks. *IEEE Access*, 7, 97505–97514.
53. Yu, Z., Hu, J., Min, G., Wang, Z., Miao, W., & Li, S. (2021). Privacy-preserving federated deep learning for cooperative hierarchical caching in fog computing. *IEEE Internet of Things Journal*, 9, 22246.
54. Sharifi, A. M., Amirgholipour, S. K., & Pourebrahimi, A. (2015). Intrusion detection based on joint of k-means and knn. *Journal of Convergence Information Technology*, 10(5), 42.
55. Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*, 5, 21954–21961.
56. Tang, F., Mao, B., Fadlullah, Z. M., Kato, N., Akashi, O., Inoue, T., & Mizutani, K. (2017). On removing routing protocol from future wireless networks: A real-time deep learning approach for intelligent traffic control. *IEEE Wireless Communications*, 25(1), 154–160.

57. Liu, W. X., Cai, J., Chen, Q. C., & Wang, Y. (2021). DRL-R: Deep reinforcement learning approach for intelligent routing in software-defined data-center networks. *Journal of Network and Computer Applications*, 177, 102865.
58. <https://www.firstpoint-mg.com/blog/top-4-challenges-in-iot-data-collection-and-management/>
59. Liyanage, M., Pham, Q. V., Dev, K., Bhattacharya, S., Maddikunta, P. K. R., Gadekallu, T. R., & Yenduri, G. (2022). A survey on Zero touch network and Service (ZSM) Management for 5G and beyond networks. *Journal of Network and Computer Applications*, 103362.
60. Gündüz, D., de Kerret, P., Sidiropoulos, N. D., Gesbert, D., Murthy, C. R., & van der Schaar, M. (2019). Machine learning in the air. *IEEE Journal on Selected Areas in Communications*, 37(10), 2184–2199.
61. Chen, M., Semiari, O., Saad, W., Liu, X., & Yin, C. (2019). Federated echo state learning for minimizing breaks in presence in wireless virtual reality networks. *IEEE Transactions on Wireless Communications*, 19(1), 177–191.
62. Hoymann, C., Astely, D., Stattin, M., Wikstrom, G., Cheng, J. F., Høglund, A., et al. (2016). LTE release 14 outlook. *IEEE Communications Magazine*, 54(6), 44–49.
63. Pramanik, P. K. D., Solanki, A., Debnath, A., Nayyar, A., El-Sappagh, S., & Kwak, K. S. (2020). Advancing modern healthcare with nanotechnology, nanobiosensors, and Internet of nano things: Taxonomies, applications, architecture, and challenges. *IEEE Access*, 8, 65230–65266.

A Comprehensive Survey on Network Resource Management in SDN Enabled Data Centre Network



Ashish Sharma, Sanjiv Tokekar, and Sunita Varma

Abstract Network Resource Management in the software-defined data centre is achieved through some intelligent software which provides the automated allocation of the resources on the basis of the instantaneous network parameters. In this paper, we present a comprehensive survey of the most relevant research on network resource management in software-defined networking (SDN)-enabled data centre networks. SDN-based resource management can optimize multiple resources in a data centre network. We begin by outlining the SDN and DCN architectures. The survey conducted by the researchers is then discussed in detail. Though there are few survey articles written on resource management there is no detailed and systematic review conducted on resource management challenges and opportunities. The recent growth of an intelligent data centre is confronted with many challenges like scalability, performance, security and energy efficiency. We surveyed the state-of-the-art resource management techniques with advantages and limitations. Cost reduction, power optimization, increased efficiency, and service stability are just a few of the many benefits of data centre resource management. Resource management methods are not entirely dynamic and do not support an unstable network, which is the biggest drawback. This review will aid the researchers in this field to identify the pros and cons associated with each resource management strategy. It will also help them to identify the best suited technique for load distribution in a specific scenario of SDN. The review of the existing literature is conducted on the basis of various parameters like load distribution capability, network efficiency, resource management, computation complexity, static or dynamic controller distribution, etc. These parameters play an immensely vital role in deciding the overall SDN performance in data centre context.

A. Sharma (✉)
Government Women's Polytechnic College, Indore, India

S. Tokekar
IET DAVV, Indore, India

S. Varma
S.G.S.I.T.S, Indore, India

Keywords Software define network · Data Centre network · Service level agreements · Resource management

1 Introduction

The structure of today's modern civilization is based on several sorts of networks. We are already able to observe the repercussions of these shifts in our day-to-day lives, manifesting themselves in spheres as diverse as the economy, technology, and politics. Networks are utilised in all of these ways, and as a result, they play an increasingly important role in the reduction of costs and the enhancement of production. The administration of our wellness and physical fitness has also become increasingly reliant on networks. The exponential growth in scope of today's networks has resulted in an increase in the number and severity of the challenges they face. Complexity and the expense of operation are two of the most significant issues faced by modern networks. We have pushed the limits of traditional networking by implementing low-power, low-capability nodes to build sensor networks. These networks are seen as a very cost-efficient alternative for data collection across a variety of applications, and we have used them to test the boundaries of traditional networking. Because the system does not have a global state or a global time, these nodes frequently function asynchronously, which further complicates the situation. In order to solve these issues, software-defined networks, often known as SDN, were developed. By noting that "In the SDN design, the control and data planes are separated, network intelligence and state are conceptually centralised, and the underlying network infrastructure is abstracted from the applications," the Open Network Foundation defined SDN. In essence, SDNs give users the power to programmatically alter the underlying network's capabilities while causing the least amount of interruption to the applications that depend on the networking infrastructure.

The main focus of SDN is on the separation of the control and data planes, the centralization of the management perspective, the open interfaces between the various planes and vendors, and the programmability of the network and its applications. SDN adds a number of features. The objective is to further separate applications from routers and switches. Point-to-point (P2P) connections between network nodes were the main focus of SDN's architecture for wired networks. Influenced by the next generation of the digital economy, new business models have flourished, paving the way for the meteoric rise of the Internet of ultra-large-scale data centres, which rely on cutting-edge technologies such as 5G, the Internet of Things, cloud computing, big data, artificial intelligence, and similar technologies to support these new services. The data centre is an infrastructure that consists of servers, storage devices, network devices (switch, router, cables, and firewall), a power distribution system, and many more. These infrastructures are interconnected through network links and switches, composed of SDN.

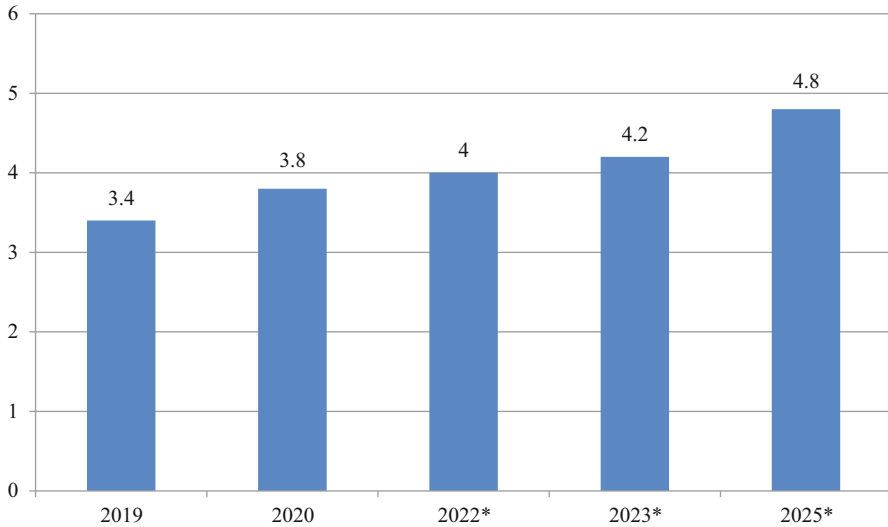


Fig. 1 Value of data centre market investment, India, in USD Billion

In recent years, a data centre has become a vital component of the new internet world [1]. The data centre is an infrastructure that consists of servers, storage devices, network devices (switch, router, cables, and firewall), a power distribution system, and many more. These infrastructures are interconnected through network links and switches, composed of SDN. In recent years, a data centre has become a vital component of the new internet world [1]. The core component of the data centre is Network Infrastructure, Storage Infrastructure, and Computing Resource. The core component of the data centre is Network Infrastructure, Storage Infrastructure, and Computing Resource. According to a report that was just released by NASCOM, the value of the data centre market investment in 2019 was USD 3.4 billion, and it is expected to reach USD 4.8 billion by the year 2025 as can be seen in Fig. 1 below. Between the years 2019 and 2025, it is anticipated that the market for data centre network infrastructure will expand at a compound annual growth rate (CAGR) of 5.5%.

Enterprise DC, Managed Service DC, Colocation DC, and Cloud DC are the four basic forms of DC. The three-layer approach monitors, analyses, and automates all types of DC resource management. We are interested in Cloud DC in this study [2]. Computing as a fifth utility has been made possible by the advent of the “cloud DC,” which allows users to access software and IT infrastructure from anywhere. In the cloud, data centre resource management is still a complex issue that is highly dependent on load on the application. Specific physical devices were used to link the applications. Cloud computing servers in traditional cloud computing environments such as data centres consequently, these servers were

frequently overprovisioned to handle issues related to maximum workload. Wasted resources and energy are the result of this. In terms of resource management, the data centre was prohibitively expensive to run. Dynamically requested services cannot be handled by the Internet's traditional network structure. Fortunately, a solution to this problem has emerged in the form of software defined networking (SDN). SDN has created numerous opportunities for computing and networking researchers in the cloud data centre. The separation of the control and data plans is, as we all know, the most important feature of SDN. The Data plane is in charge of routing and managing packets between the source and destination ports. To operate the packet forwarding mechanism, the control plane is software written in a controller-compatible programming language. The data centre network (DCN), which is made possible by software-defined networking, serves a unique role in a data centre by connecting all network resources. More than that, managing resources is a difficult task because many issues are interconnected, such as resource heterogeneity, asymmetric communication, inconsistent workload and dependency on resources. Additionally, therefore, this study examines the contributions of previous research to SDN-based cloud DCs in terms of network performance and energy efficiency. The major tasks of DCN resource management include improving performance, efficiency, and lowering operational costs [3, 4].

The purposes of our participation in this survey are as follows:

1. We analyze the taxonomy of current trends in resource management strategies, emphasizing their advantages and disadvantages.
2. We discuss the Trend and Opportunities in SDN
3. We identify future research works, which constitute the basis of present and future research recommendations.

This paper's contribution is an analysis of the methods proposed in the literature for managing network resources in SDN-enabled data centre networks. To categorize the resource management options in the SDN-based network including DCN & cloud, Wireless, and WAN, we conduct a literature review of papers published by Springer, IEEE, Elsevier, and ACM between 2015 and 2022. Papers from Springer, IEEE, Elsevier, and ACM, as well as other papers relating to SDN load balancing improvement solutions, are used to supplement this work; their distribution is shown in Fig. 2. SDN, Resource Management, and Data Centre are all terms that are searched for simultaneously. Researchers can use the pie chart in Fig. 2 to pinpoint ACM, Elsevier, IEEE, and Springer journals covering the topic of resource management in SDN.

The remainder of the paper is laid out as follows. In Sect. 2, we discuss the history of SDN and DCN. The researcher's relevant work is presented in Sect. 3. SDN-DCN resource management challenges are covered in Sect. 4. Section 5 depict SDN-prospective DCN's trends and opportunities. The paper came to a close with Sect. 6 which is Conclusion and Future Work.

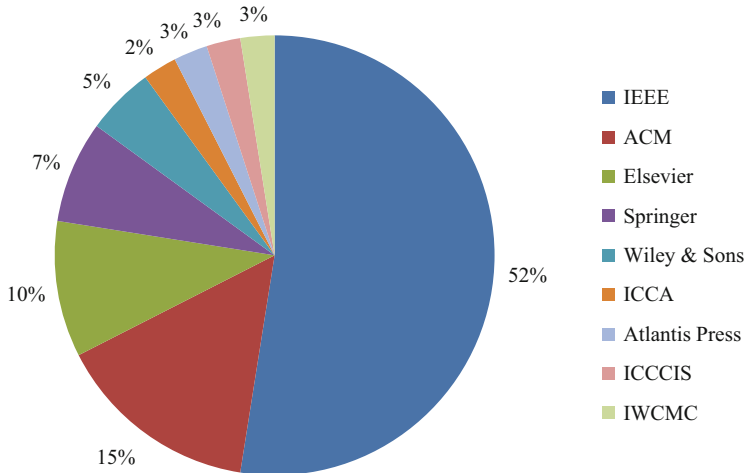


Fig. 2 Percentage of reviewed paper in publication

2 Background

2.1 A Subsection Sample

The separation of the data (also known as forwarding) and control planes is suggested by the potential network design known as software-defined networking (SDN). To coordinate the network, it uses a logically centralised controller supported by a network-wide global view [106]. By switching from network administration to network programming, it fosters creativity. SDN, in theory, aims to solve the legacy networks’ manageability issues by converting static networks into dynamically-programmed ones. In computing and networking, the SDN architecture defines how a system can be designed to use a combination of open, software-based technologies and commodity networking hardware. Open Interface is a fundamental property of SDN. Which is use to connect the network resources and the network traffic. This link is controlled by software that is being developed in response to changing needs. SDN architecture separates the control and data planes of the network stack. SDN architecture comprised a three-layer Application layer, Control layer, and Infrastructure layer, as shown in Fig. 3 [5].

The Infrastructure Layer (Data plane) The data plane, also called the forwarding plane, is present in the hardware of switches and is accessible via the software. Although administrators have some control over the data plane, that control is limited. Because SDN uncouples the data plane and control plane, OpenFlow is needed as a communication route for coordinating these two layers [3]. These planes build routing tables on the routers and switches to identify which packets should travel from A to B and which should go to C. Despite their uniqueness, conven-

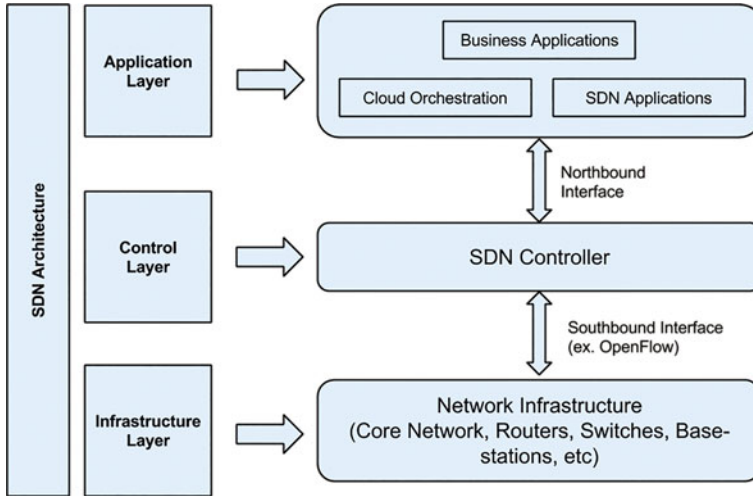


Fig. 3 The traditional SDN architecture [43]

tional routers and switches nonetheless follow IEEE standards and incorporate all necessary planes into their firmware. It comprises multiple interconnected network elements like simple switches, routers, and base stations. The Infrastructure layer connects with SDN southbound interface (or southbound APIs) to the Control layer (Control Plan). However, these APIs are tightly bound to the forwarding elements of the physical infrastructure.

The Control Layer (Control plane) The Control plane transcends the boundaries of a conventional software layer in its entirety. On this level, we have a face-to-face encounter with a concrete member of the hierarchy: the controller. The additional information presented here illustrates how the axes are separated from one another. This plane symbolises the conceptual centre of the argument. One decision point made up the entirety of the control plane's simplest iteration. As a consequence of this, the method makes the process of designing, programming, and resolving logical contradictions simpler. This architecture, on the other hand, is extremely vulnerable to a failure at a single point that might have devastating consequences. Problems with scalability arise when the processing capability of the controller needs to scale up proportionally more as the network increases.

Researchers came up with the idea of physically and conceptually decentralising the SDN's Control plane. To do this, they proposed the concepts of a control hierarchy composed of main controllers and subsidiary controllers as a way to decentralise control. This was done in an effort to mitigate the negative effects of centralised controllers. As a consequence of this, the network was able to divide the load among a number of different physical controllers, hence lowering the probability of experiencing bottlenecks. Comprises of a centralized SDN controller where the network administration takes place. It is responsible for the network

status monitoring, link discovery, policy generation & development with forwarding table management. The Control layer interacts with the application layer through Northbound Interfaces (or northbound APIs).

The Application Layer (application plane) could include network applications and services such as load balancing, routing, security, mobility & wireless, etc., are implemented. The main strength of SDN architecture is that it provides an abstracted holistic view of the entire network. Attributes, services, and rules are all specified at the application layer, which is also the name of this layer. Applications have to have a solid understanding of the infrastructure of the network in order for them to respond effectively. These apps are able to develop capabilities that span end to end and can respond to changes in the underlying network. Applications are able to dynamically adapt the behaviour of the network in order to meet shifts in the network's topology, feature needs, or policy needs. The application programming interfaces (APIs) are what make the previously described layers dependent on one another (APIs). We make the network more intelligent by integrating real-time networking information with the application.

Northbound APIs The programming interface between SDN controllers and the network applications built on top of them is called a Northbound Interface (NBI) API. Network applications can communicate with controllers through a northbound interface and call the services that the controllers make available and that the applications need for proper functioning, such as the global network view. We believe there are a few northbound APIs, but none of them is the industry standard. An ONF project has been launched with the goal of forming a working group for standardising NBIs.

Southbound APIs The interface that enables SDN controllers to talk to switches in order to manage and observe their functioning is known as a Southbound Interface (SBI) API. SBIs, as opposed to NBIs, are standardised to allow the management of network devices from several vendors. OpenFlow [42], which is maintained by the ONF, is the de facto SBI standard at the moment. Other SBIs exist, such ForCES, which was suggested by the IETF. Either in-band or out-of-band connections can be used with SBIs. In-band control includes using the same network (physical connections) to send the controller-switch communication as data traffic. A separate network is used for control and data traffic in out-of-band control. Most SDN deployments choose out-of-band control due to reliability concerns, however in-band control may be preferable due to its financial advantages.

2.2 *Datacentre Architecture*

In order to support the next-generation Computing-as-a-Service (CaaS) and Cloud computing infrastructures, data centres have advanced from using mainframe systems and enterprise networks to sophisticated networks of 100,000 or more

servers. The proliferation of data centres has resulted in a proportional rise in the amount of energy consumed by data centres as well as an increase in the amount of network traffic experienced by data centres. As a result, there is a considerable push for novel study of data centres, with the ultimate goal of developing effective data centre network (DCN) architectures and ways for substantially reducing energy consumption. A data centre is a collection of computing, storage, and networking resources that are connected via a communication network. In a data centre, the Data Centre Network (DCN) is crucial since it connects all of the data centre resources. To satisfy the expanding demands of Cloud computing, DCNs must be scalable and efficient enough to connect tens of thousands, if not millions, of servers. The data centre network design is based on a tried-and-true layered approach that has been tested and refined over the course of several years in some of the world's largest data centre deployments. The layered approach is the fundamental building block of data centre design, and it aims to improve scalability, performance, flexibility, resiliency, and maintenance while also reducing costs. The Data Centre is typically a three-tiered hierarchical internetworking concept. As illustrated in Fig. 2, the model is made up of three layers: access layer, aggregation layer, and core layer [6].

Core Layer Data centres can easily connect to the internet via core layer switches. In and outbound data packets are routed through the data centre's core switch. The Layer 3 routing flexibility of the core switch is complemented by its lightning-fast throughput. This capability enables high-speed packet switching on the data centre's backplane for both incoming and outgoing traffic. The primary function of this layer is to supply a fault-tolerant, fully redundant Layer 3 routed fabric. The core layer also communicates with several aggregation subsystems. Traffic between the campus core and the aggregation layer can be load-balanced with the help of Cisco or other Express Forwarding-based hashing techniques while an interior routing protocol such as OSPF or EIGRP is being run on the core layer.

Aggregation Layer Core layer switches connect the aggregation layer switches to each other. Service modules, Layer 2 domain definitions and spanning tree processing are just some of the important functions that these modules provide. An active-passive high availability mode is used for the aggregation layer. Layer 2 domain definitions, spanning tree processing, and default gateway redundancy are some of the key features that it contributes. Optimizing and securing programmes for multi-tier traffic across servers can be accomplished through the utilisation of services such as server load balancing and firewalls. In Fig. 4, the integrated service modules are represented by the smaller symbols that may be found inside the aggregation layer switch. These modules have the capability of providing a wide variety of services, including but not limited to content switching, firewall defence, SSL offloading, intrusion detection, network analysis, and more. At the aggregation switch, a network's Layer 2 and Layer 3 segments are clearly separated. This layer is also called the Distribution layer.

Access Layer Data centre servers are physically connected to the access switch. All of the servers in a network are connected at the access layer. Various types of

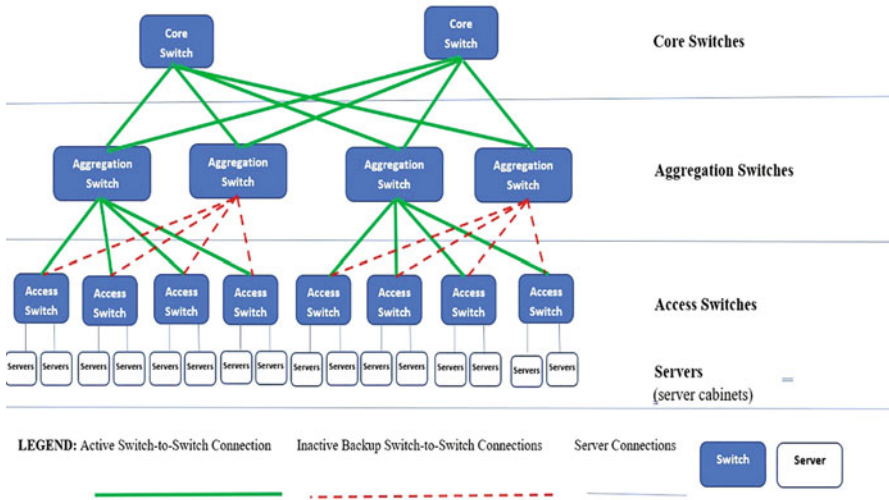


Fig. 4 The traditional data centre architecture [42]

servers, such as standalone machines, blade servers with built-in switches, pass-through blade servers, clustered standalone machines, and mainframes with Open Systems Architecture (OSA) adapters, are included. The access layer of a network might be composed of embedded blade server switches, modular switches with a fixed configuration of 1–2 rack units (RU), or a combination of the three. Switches offer link layer and network layer topologies, making it possible for them to accommodate a wide range of managerial and broadcast domain requirements from servers. Access switches are usually located at the top of the rack. Multiple access layer switches are interconnected with the aggregation switch.

2.3 Data Centre Design Models

Multi-Tier Model A cluster-based multi-tier data-centre has key differences from high-performance computing systems. Data-centres must run a variety of server software, each with its own hardware and software requirements. High-performance computing systems use multiple copies of a programme to complete a task in parallel. Proxy, application/web, and database layers make up a multi-tier data centre’s standard trinity. Due to the different needs and behaviours of each tier, it is difficult to analyse architectural components such as file system, I/O, network protocol, etc. and their impact on a multi-tier data centre. As illustrated in Fig. 5 multi-tier architecture abstracts various functionalities, which is useful as dynamic web content grows. I/O interconnect technologies are crucial for inter- and intra-cluster communication in multi-tier data centres. Distributed web servers improve

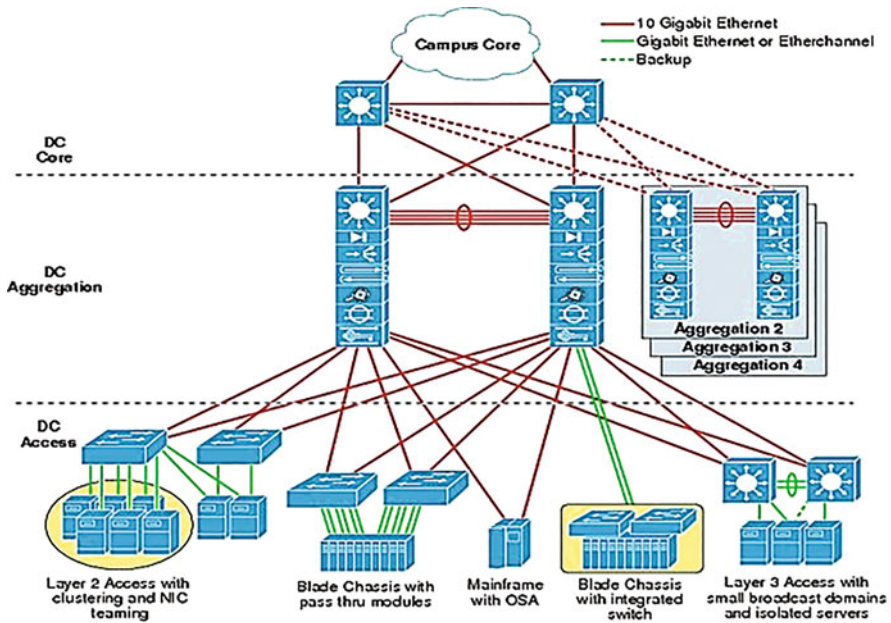


Fig. 5 Data centre multi-tier model topology

throughput and response time. Enterprise resource planning (ERP) and customer relationship management (CRM) systems are supported by a foundation of layered web, application, and database design. This style of design is compatible with a wide variety of web service architectures, such as those based on Microsoft.NET or Java 2 Enterprise Edition. ERP and CRM software like Siebel and Oracle use web service application environments. It is crucial to the multi-tier architecture that services for networked security and application optimization are made available.

Server Cluster Model High availability, load balancing, and more computational power are just a few of the reasons why clusters of servers are used in today's data centre environment. Clusters are large deployment units made up of tens or hundreds of individual server cabinets connected via large, high-radix cluster switches and outfitted with top-of-rack (TOR) switches. Each cluster is designed to make multiple central processing units (CPUs) function as if they were part of a single, unified; high-performance system by means of specialized software and high-speed network interconnects. Historically, server clusters have been utilized in the context of research at academic institutions, scientific laboratories, and the military for specialized applications. The widespread adoption of server clusters in businesses is a direct result of the widespread application of clustering technology to a growing variety of applications. Recent years have seen the concept of a server cluster spread from academic institutions to large-scale industries like banking, manufacturing, and the media. The concept of a server cluster is useful not only

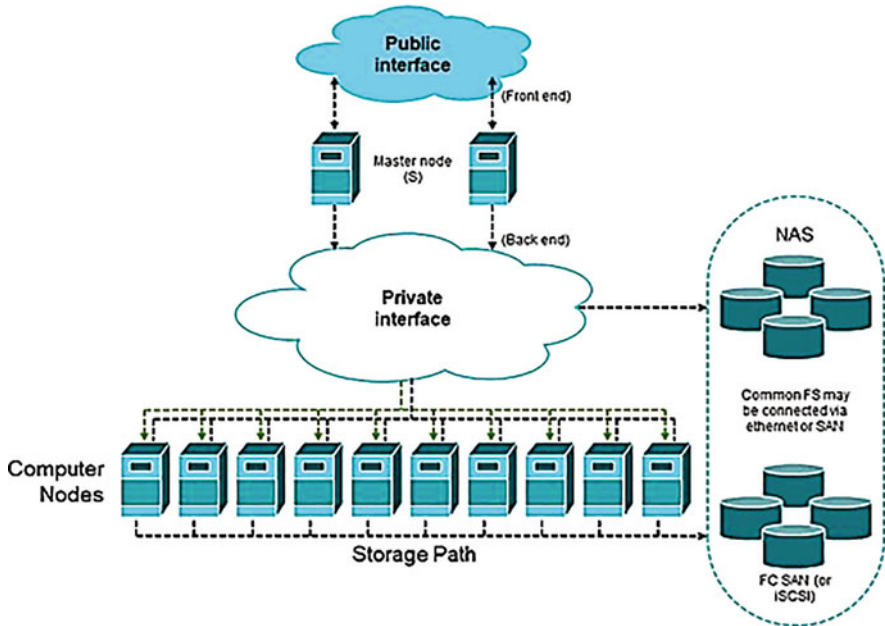


Fig. 6 Server cluster model of data centre layout

for grid and utility computing but also for HPC, parallel computing, and high-throughput computing. These designs as shown in the Fig. 6 are typically based on specialized application architectures created to meet a wide variety of niche market needs in the business world.

3 Related Work

This section presents the variety of important and significant solutions offered by the researchers over the last decade. The objective of discussing the existing works in the field of resource management and load distribution in SDN framework is to present the spectrum of solutions available to attain the most optimal solution. Open research challenges in modern cloud DC systems include resource management and load balancing, as well as issues of security and privacy. This section presents a detailed analysis of some of the relevant previous works related to this resource management of SDN. The potential of Network Resource Management in Data Centre Network with SDN has been recognized and explored by many researched over the last decade.

In a dense network, Yang et al. [7] provided a solution for the video streaming problem and suggested that in order to overcome the problem, video layer selection

and resource allocation must be done correctly. They have applied Lyapunov optimization approach to decompose the problem of resource allocation into two level problem viz. wireless resource allocation and the respective video layer selection. An effective routing scheme for the video streams was also presented by the researchers to utilize the small cell base stations interaction.

Ndikumana et al. [8] proposed the multi-access edge computing concept for reducing the network delay and alleviate the load on the data centre. They proposed an idea of a 4c framework for mobile edge computing. A distributed optimization control algorithm is developed by researchers, which increases bandwidth saving and minimized delay. By collaborative communication computation, the MEC server's caching and control model resource allocation can be done.

Karmoshi et al. [9] developed an application-aware resource method named as virtual physical switch software-defined network (VPS-SDN). This model proposed a novel structure for network virtualization for the multitenant cloud data centre. This scheme achieves high network utilization.

Yahya et al. [10] have suggested that large-size topologies consumed more resources than the small-size topologies in their research work. The number of sources depends upon the type of switches and controller. By using Open v Switch, CPU utilization is more as compared to another switch. As per their study, the CPU utilization of OVS controller was found to be inferior as compared with another controller.

To improve end-to-end system performance and meet different applications' requirements, Chen et al. [11] developed a joint caching computer strategy. They have also proposed a solution to the server selection problem. Energy cost and network usage cost are also minimizing through resource allocation problem solutions.

Chen et al. [12] studied different resource allocation issues by joining networking caching and computing. By balancing the server's load and reducing the network usage, they develop a framework for improving the performance. A discrete stochastic approximation (DSA) algorithm is developed.

Tso et al. [13] surveyed various resource management strategies for the network and server. In their research, they emphasized over the necessity and limitations of adaptive resource provisioning. They have also analysed the challenges and opportunities for adaptive and measurement-based resource allocation. As per their findings, more matrices are required to developing robust resource measurements and better trade-off between network-wide stability and adaptively.

Braiki et al. [14] categorize their survey for resource management issues based on use method, approach and objective model. For better performance, multi-objective models are better in terms of energy consumption and resource utilization. They used a cloudsim simulator. In their proposed method, average energy consumption improves by 12% and average resource utilization by 15%.

Cao et al. [35] reviewed modern data centres, resource management issues. The first issue is figuring out how to combine diverse resources (hardware and software) into a uniform platform (virtual resource). The second is the problem in figuring out how to manage the numerous resources effectively in the organisation. The third

issue is that of resource services in particular, network services, selecting a suitable method of resource management among several resources' platforms. As a result, it is challenging and complex the following conditions should be considered: resource accessibility management, a temporary storage pool, and the ability to be flexible executing network architectures (such as resource allocation).

4 Challenges Associated with the Resource Management in SDN-DCN

Technological changes in the data centre have not delivered their full potential in boosting productivity and economic growth. There are several challenges to managing these technological changes in SDN-DCN. This section discusses some of the difficulties related to SDN-DCN resource management. The comprehensive objective of a DCN is to ensure that it endlessly meets the SLA of the application it hosts, with minimize underutilized and avoiding over-utilized total resources. In DC, resources are diversified and distributed. To manage these resources is a complex system. The major challenge is building an intelligent autonomous resource management system that can self-alleviate and self-heal from any system failures [15].

When in DCN any requests come for one or more resources, the data centre network provides master controller, it schedules the available resources by making them public. Scalability, performance, security, reliability, and energy efficiency are just a few of the resource management concerns that must be addressed [16]

1. **Scalability:** Every day, data centre operators face the challenge of scaling their operations and acquiring enough hardware to meet the demands of complex IT systems (such as increasing compute, storage and network needs). The Internet of Things (IoT), social media platforms, on-demand video, and the global digital revolution are all contributing to a rise in computer density, which in turn is increasing the need for scalable data centres. There has to be systems and strategies in place to manage this expansion that are cost-effective and able to scale. It is one of the most challenging aspects of SDN-DCN. Because of the SDN controller, DC now has the ability to scale up to hundreds of thousands of network resources. It is preferable to create a controller that can handle big dynamic flow tables when dealing with a high number of resources. Designing an intelligent, scalable controller for controlling resources in SDN-enabled data centres requires more research [17]. To meet the tremendous growth in compute power and storage, data centre operators must expand crucial infrastructure (power and cooling) as well as physical space. The reliability of the current operating system must not be compromised in order for MTDC operators to offer electricity and cooling infrastructure promptly, efficiently, and at the lowest cost.
2. **Performance:** The performance guarantee is an essential issue between the data centre service providers and users. User satisfaction and DC cost are

directly linked to performance management. To improve the performance of the DC resource management, the researcher has to study more key performance indicators (KPI) [18]. Data centres typically monitor CPU utilisation as a proxy for performance. However, if application demand remains stable, can we cut the number of machines in the data centre in half? No, probably not at all. This assumes that the average CPU utilisation per machine in the data centre is 50%. Because of the wide variety of workloads and hardware in use in data centres, it can be challenging to conduct a comprehensive performance analysis of the infrastructure's effectiveness. Emulation, automation, and analytics must be utilised by providers at every stage of the delivery process, beginning with design and construction and continuing through deployment, operation, and optimization. Only then can providers guarantee that customer expectations will be satisfied at each stage.

3. **Security:** In SDN-enabled data centre security, associated problems are a significant concern. Effective security must be created and placed at the application plane in SDN for data centre network protection. The controller is the primary target of threats in the SDN-DCN. The attacker targets the controller for serious damage to the data centre network. For the unauthorized access, several alleviate plans of action would be developed. Which reduces the threat and provides high-level security to protect the data centre network. Despite the fact that data centres may be quite complicated, they are only required to adhere to a single comprehensive security policy; yet, each individual component of that policy needs to be carefully evaluated. The term "software security" refers to one category of protection, while "physical security" refers to another. The phrase "physical security" is used to refer to a wide range of different strategies and approaches that are utilised to prevent unwelcome access by individuals from the outside. By using software or virtual security measures, the network is secured against potential invaders who would otherwise be able to circumvent firewalls, crack passwords, or gain access through other security weaknesses. Intruders may be able to do any of these things [19].
4. **Reliability:** It is essential to ensure high reliability in an SDN-enabled data centre network. It is an essential factor in communication within the data centre network. It is very challenging to develop the intelligent and validated SDN controller that manages the network to increase network availability. When any controller fails or is over-utilized, the system must drive through the available alternative controller to maintain a reasonable level of reliability. Reliability-aware capabilities of the controller can improve the reliability of the controller plane. To meet the service level agreement (SLA), SDN enabled DC is more reliable. It is absolutely necessary to have a reliable network in the data centre in order to construct and operate online services that are highly available and scalable. Even while there is a lot of monitoring going on at the device and link levels, the full implications of how stable the network infrastructure is for the software systems that rely on it are still a mystery. The fundamental difficulty lies in identifying the connection that exists between the effect of the software system and problems at the device and link levels. To begin, the redundancy that is built

into the infrastructure of most networks ensures that the majority of network outages do not cause problems with the software systems (including redundant devices, routes, and protocols). Second, automated repair systems are frequently employed in the infrastructure of large-scale networks so that problems can be fixed as soon as they are identified. This makes it possible to resolve issues in a timely manner [20].

- 5. Energy Efficiency:** In the recent growth of the Information and Communication Technology (ICT) sector, energy efficiency poses a serious challenge. Data centres in the United States and elsewhere use between 1% and 3% of the world’s total electricity, according to various reports. With the proliferation of IoT devices and AI, these figures might presumably rise dramatically. These days, data centres are an integral part of any discussion on computing or networking. Data centres house computers and networks that collect, organise, and make available vast amounts of information. About \$20 billion is spent annually on their development, and they generate nearly as much carbon dioxide as the airline sector does. The energy usage of network resources within a data centre is predicted to climb to roughly 50% in the next few years in DC. The major challenge is increasing the performance DC uses more redundant resources, which reduces the energy efficiency level. Another challenge is minimizing the underutilization and avoiding over utilized resources for better energy efficiency without disrupting the SDN-enabled DC operation [21].

In this section, a side-by-side comparative analysis of the selected technique is performed in terms of the advantages and limitations as shown in Table 1.

Table 1 Resource Management (RM) Methods and their comparative analysis

Reference	Technique for RM	Advantages	Limitation
[22]	Reinforcement learning	1. Proposed method decreases the SLA (service level agreements) violation time 2. To serve more VM schedule request 3. Migration between resource are decreased	1. More power consumption 2. More dynamic testing environment
[23]	Scalable routing and resource management model (SRRM)	1. Load balancing 2. High degree of scalability	1. Dynamic load balancing 2. Energy efficiency
[24]	Dynamic resource management algorithm (DRMA) Bin packing algorithm	1. Minimize underutilized resources 2. Avoid over-utilized resources	1. Negative impact on the performance 2. Additional energy required
[25]	Resource allocation scheme	1. Less packet loss rate 2. Improvement in link utilization.	1. Congestion management not implemented 2. Only theoretical approach

(continued)

Table 1 (continued)

Reference	Technique for RM	Advantages	Limitation
[26]	Green resource management techniques	1. Decreased the energy consumption 2. Decreased the carbon dioxide emission 3. Decreased dynamic cost	1. More components can handle like green based cloud balancing 2. Green-based match making
[27]	Hybrid virtual machine consolidation approach Beam search algorithm	1. Minimization of energy consumption and SLA violation	1. Workload predicting 2. Reduce the VM migrations
[28]	Traffic analysis-based energy-saving approach	1. Energy efficient 2. Reduce round trip delay 3. More bandwidth utilization	1. Traffic loss is possible 2. Only static resource management
[29]	VPS-SDN (virtual physical switch software defined network)	1. High network utilization 2. Scalable 3. More QoS	1. Only prototype model 2. For less network size
[30]	Overhead reduction scheme	1. Less network latency 2. More throughput 3. More network performance	1. Only for small DCN topology 2. More network scenarios
[32]	Network topology framework	1. Reducing complexities 2. Improving resource administration	1. More VM overheads 2. VM placement issues 3. VM allocation related problems
[33]	CPU-based and CPU memory-based techniques	1. Achieve higher resource utilization and more reliability 2. Shorter response time 3. Higher throughput and transaction rate	1. Only for small network 2. Only theoretical approach
[34]	Chopin framework (an intent-driven framework)	1. Better resource efficiency 2. Better resource allocation	1. Not support for unstable networks
[36]	Intelligent network resource allocation scheme	1. Avoiding congestion and idleness. 2. Automatically assignment of network resource	1. Impact to delay and jitter are not considered 2. Dynamic queue mapping is not done

5 Promising Trend and Opportunities in SDN-DCN

Data Centre Network with SDN-enabled has a new paradigm. Dynamic programming of the controller and integration of the data centre controller with the SDN controller enables optimised network resource management. It also improves network scalability and manageability. Monitoring or predicting the traffic by the controller is also one more advantage of the SDN-enabled data centre network.

The distinguishing characteristics of SDN create more opportunities for network resource management in DCN. The simultaneous use of server resources and network resources can bring more innovations by integration and clustering of DC controller with SDN controller. SDN intelligent controller and DCN controller integration bring DC network resource management innovation [31].

Year-over-year changes in SDN that take DC networking to the next level. There are a few trends in SDN-DCN.

1. **Edge Computing** – In 2022 (and beyond), edge computing will be gaining high importance in the DC network. The demand for edge computing will grow because people have adopted more intelligent technologies in their homes and businesses. In parallel IoT market is also overgrowing. With this growing demand for reliability, speed and connectivity will be managed by edge computing. Edge computing's future is rapidly arriving. There is endless potential to change the world as processors get stronger, storage gets more affordable, and networks access gets better. Edge computing will advance in the future along with cutting-edge networks like 5G, satellite mesh, and artificial intelligence. You've now unlocked some very far-reaching opportunities by having more capacity and power, better access to swift and wider networks (5G, satellite), and smarter computer-based machines (AI) [37].
2. **Green Computing** – On coming year, sustainability issues like water usage, energy emissions, and consumption is growing concerns in the DC network. Renewable resources and management are a big focusing area at the DC network. Look for the DC to find ways to reduce their impact on the environment and help other sectors. The term "green computing," sometimes known as "green IT" (Information Technology) or "green technology," refers to the environmentally responsible use of computing systems and the resources they necessitate. Environmentally friendly computing refers to efforts to develop, deploy, and retire computer systems and components with little impact on the natural world. Some examples of these practises include designing energy-efficient computing equipment, implementing energy-efficient processing units and servers, reducing the use of harmful chemicals, advocating for the recyclable nature of digital products, and disposing of electronic waste in an environmentally responsible manner (e-waste). Green computing strives perpetually to make computing less harmful to the environment. Green Computing was originally called Energy Star when it was introduced in 1992 [38].
3. **Automation** – Due to the coronavirus pandemic, more automation is another aspect of network resource management in SDN-enabled DC networks. More DC shifts to remote monitoring capabilities and routine services like updating and patching to limit contact with other people. Skilled staffing issues are still a concern in DC. The potential for edge computing to automate many existing DC processes is substantial. Artificial intelligence (AI) and robots have progressed so quickly that they have pushed the limits of automation. These days, robots can do a lot of work with hardly any help from humans. Technological automation is not only eliminating repetitive tasks, but also vastly improving workers'

talents. More than half of human labour might be replaced by automated robots, according to some estimates. Automation is utilised in numerous industries, including manufacturing and banking, to improve efficiency, security, profits, and product quality. Automation will enhance reliability and connectivity in a highly competitive market. It seems that in the future, everything will be easily accessible thanks to automation [39].

4. **5G Technology** – DC resource management will require considerable changes to accommodate 5G technology. Primary DC network resources and requirements like QoE, Network inter-operability, performance is resolved by SDN-enabled 5G network. It is mandatory for a data centre to host and stream data at significantly higher speeds, volumes, and lower latencies. Due to better efficiency and bandwidth, 5G network enhances resource management efficiency and can serve faster in the DC network. The next leap forward in wireless technology will be made possible by fifth-generation wireless technologies (5G). There will be more storage space, higher transfer rates, and less lag time with this new technology. 5G has enormous potential for facilitating developments across many sectors, including those related to public safety, transportation, and healthcare. In the emerging IoT ecosystem, where gadgets are increasingly connected to one another, it will also have an impact on sustainability [40].
5. **Hyperscale** – Big data and cloud computing environments are usually used in Hyperscale computing. DC conventional computing architecture structural design is often different from hyper-scale computing architecture. Hyperscale data centre, market size, networking equipment value grows at around 30% from 2018 to 2020. Open architecture, edge computing, and security potentially affect respondents in hyper-scale DC. Massive in size, hyper-scale data centres house thousands of servers, racks of networking hardware, tonnes of cooling and power infrastructure, and more. Demand for data centres has increased as Covid-19 has spread around the world, with strong purchases coming from hyperscale businesses and cloud platforms while spending from many enterprise users has slowed [41].

6 Conclusion and Future Work

Over the last few years, researchers have become increasingly interested in data centre networks that use software-defined networking (SDN). The challenges and opportunities associated with the various techniques for resource management in an SDN-enabled data centre network are thoroughly covered in this paper. Also included are a number of different resource management approaches, as well as their advantages and disadvantages. The bulk of procedures are restricted in their ability to be automated and have low energy efficiency. This paper presents a wider spectrum of load distribution and resource allocation techniques proposed by various researchers over the last decade. The advantages and limitations of these techniques can help the readers to identify the best possible technique for the

scenario under consideration in their research. We then talked about the most recent software-defined network data centre networks trends. A major focus of future research will be on assessing the failure of resources in conjunction with intelligent controller resource management in SDN-DCN. However, there are various concerns and obstacles that must be addressed to enhance the efficiency, reliability, and cost optimization of the data centre network overall.

References

1. Kreutz, D., Ramos, F., Veríssimo, P. E., Esteve, C., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1), 14–76.
2. https://www.cisco.com/c/en_in/solutions/data-centre-virtualization/what-is-a-data-centre.html
3. Sun, X., Ansari, N., & Wang, R. (2016). Optimizing resource utilization of a data centre. *IEEE Communications Surveys & Tutorials*, 18(4), 2822–2846.
4. Ma, J. (2016). *Resource management frameworks for virtual data centre embedding based on software define networking, computers and electrical engineering*. Elsevier Ltd.
5. Xu, Y., Sun, Z., & Sun, Z. (2017). SDN-based architecture for big data network. In *2017 IEEE international conference on cyber-enabled distributed computing and knowledge* (pp. 513–516). IEEE.
6. https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Data_Centre/DC_Infra2_5/DCInfra_1.html
7. Yang, J., Yang, B., Chen, S., Zhang, Y., Zhang, Y., & Hanzo, L. (2019). Dynamic resource allocation for streaming scalable videos in SDN aided dense small-cell networks. *IEEE Transactions on Communications*, 67(3), 2114–2129.
8. Ndikumana, A., Tran, N. H., Ho, T. M., Han, Z., Saad, W., Niyato, D., & Hong, C. S. (2020). Joint communication, computation, caching, and control in big data multi-access edge computing. *IEEE Transactions on Mobile Computing*, 19(6), 1359–1374.
9. Karmoshi, S., Shuo, W., Saleh, F., Li, J., & Zhu, M. (2019). VPS-SDN: Cloud datacentres network resource allocation. In *HP3C'19: Proceedings of the 3rd international conference on high performance compilation, computing and communications* (pp. 100–105). Association for Computing Machinery.
10. Yahya, E. B., & Al-Somaidai, M. B. (2019). Network parameters effects on system resources in software defined networks. In *ICICT'19: Proceedings of the international conference on information and communication technology* (pp. 89–95). Association for Computing Machinery.
11. Chen, Q., Yu, F. R., Huang, T., Xie, R., Liu, J., & Liu, Y. (2016). Joint resource allocation for software defined networking, caching and computing. In *2016 IEEE global communications conference (GLOBECOM)* (pp. 1–6). IEEE.
12. Chen, Q., Xie, R., Huang, T., Liu, J., & Liu, Y. (2017). Software defined networking, caching and computing resource allocation with imperfect NSI. In *GLOBECOM 2017–2017 IEEE global communications conference* (pp. 1–6). IEEE.
13. Tso, F. P., Jouet, S., & Pezaros, D. P. (2016). Network and server resource management strategies for data centre infrastructures: A survey. *Computer Networks*, 106, 209–225.
14. Braiki, K., & Youssef, H. (2019). Resource management in cloud data centres: A survey. In *15th International wireless communications & mobile computing conference (IWCMC)* (pp. 1007–1012). IEEE.

15. Saeed, N. S. B., & Alenazi, M. J. F. (2020). Utilizing SDN to deliver maximum TCP flow for data centres. In *ICISS'20: Proceedings of the 3rd international conference on information science and systems* (pp. 181–187). Association for Computing Machinery.
16. Xu, G., Yang, J., & Dai, B. (2015). Challenges and opportunities on network resource management in DCN with SDN. In *2015 IEEE international conference on Big Data (Big Data)* (pp. 1785–1790). IEEE.
17. Hossein, A., Watts, M., & Ahmadi, K. (2019). An overview of multi-controller architecture in software-defined networking. In *10th Annual conference of computing and information technology research and education New Zealand (CITREZZ2019) and the 32nd annual conference of the national Advisory committee on computing qualifications* (pp. 9–11). Hamilton, Nelson.
18. Shirmarz, A., & Ghaffari, A. (2020). Performance issues and solutions in SDN-based data centre: A survey. *The Journal of Supercomputing*, 76, 7545–7593.
19. Leng, X., Hou, K., Chen, Y., Bu, K., & Song, L. (2018). SDNKeeper: Lightweight resource protection and management system for SDN-based cloud. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)* (pp. 1–10). IEEE.
20. Abdullah, M. N., Dawood, A. S., & Taqi, A. K. (2017). Network resource management optimization for SDN based on statistical approach. *International Journal of Computer Applications*, 177(6), 5–13.
21. Bahari, H. I., & Shariff, S. S. M. (2016). Review on data centre issues and challenges: Towards the green data Centre. In *2016 6th IEEE international conference on control system, computing and engineering* (pp. 129–134). IEEE.
22. Telenyk, S., Zharikov, E., & Rolik, O. (2018). Modeling of the data centre resource management using reinforcement learning. In *2018 International scientific-practical conference problems of infocommunications. science and technology (PIC S&T)* (pp. 289–296). IEEE.
23. Celenlioglu, M. R., Tuysuz, M. F., & Mantar, H. A. (2018). An SDN-based scalable routing and resource management model for service provider networks. *International Journal of Communication System*, 31(8), e3530.
24. Ngenzi, A., Selvarani, R., & Nair, S. R. (2015). Dynamic resource management in cloud data centres for server consolidation, *arXiv:1505.00577*.
25. Thazin, N., Nwe, K. M., & Ishibashi, Y. (2019). Resource allocation scheme for SDN-Based cloud data centre network. In *Proceedings of the 17th international conference on computer applications, (ICCA)* (pp. 15–22).
26. Surendran, R., & Tamilvizhi, T. (2018). How to improve the resource utilization in cloud data Centre? In *2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)* (pp. 1–6). IEEE.
27. Zharikov, E., Telenyk, S., Rolik, O., & Serdiuk, Y. (2019). Cloud resource management with a hybrid virtual machine consolidation approach. In *2019 IEEE international conference on advanced trends in information theory (ATIT)* (pp. 289–294). IEEE.
28. Paliwal, M., & Shrimankar, D. (2019). Effective resource management in SDN enabled data centre network based on traffic demand. In *IEEE Access* (Vol. 7, pp. 69698–69706). IEEE.
29. Pranata, A. A., Jun, T. S., & Kim, D. S. (2019). Overhead reduction scheme for SDN-based data Centre networks. *Computer Standards & Interfaces*, 63, 1–15.
30. Baig, S. R., Iqbal, W., Berral, J. L., Erradi, A., & Carrera, D. (2019). Adaptive prediction models for data centre resources utilization estimation. *IEEE Transactions on Network and Service Management*, 16(4), 1681–1693.
31. Abbasi, A. A., Abbasi, A., Shamshirband, S., Chronopoulos, A. T., Persico, V., & Pescape, A. (2019). Software-defined cloud computing: A systematic review on latest trends and developments. In *IEEE Access* (Vol. 7, pp. 93294–93314). IEEE.
32. Abbasi, A. A., Shamshirband, S., Al-qaness, M. A. A., Abbasi, A., AL-Jallad, N. T., & Mosavi, A. Resource-aware network topology management framework, networking and internet architecture (cs.NI); Machine Learning (cs.LG) *arXiv:2003.00860*

33. Hamed, M. I., ElHalawany, B. M., Fouda, M. M., & Eldien, A. S. T. (2017). A novel approach for resource utilization and management in SDN. In *2017 13th International computer engineering conference (ICENCO)* (pp. 337–342). IEEE.
34. Heorhiadi, V., Chandrasekaran, S., Reiter, M. K., & Carnegie, V. S. (2018). Intent-driven composition of resource-management SDN applications. In *CoNEXT'18: Proceedings of the 14th international conference on emerging networking experiments and technologies* (pp. 86–97).
35. Cao, R., Tang, Z., Li, K., & Li, K. (2020). *Computing, storage, and networking resource management in data centres, data centre handbook: Plan, design, build, and operations of a smart data centre* (2nd ed.). Wiley.
36. Cui, H., Ma, C., Lai, W., Zheng, L., & Liu, Y. (2015). Accurate network resource allocation in SDN according to traffic demand. In *Proceedings of the 4th international conference on mechatronics, materials, chemistry and computer engineering* (Advances in Computer Science Research). Atlantis Press.
37. Balasubramanian, V., Alokaily, M., Reisslein, M., & Scaglione, A. (2021). Intelligent resource management at the edge for ubiquitous IoT: An SDN-based federated learning approach. *IEEE Network*, 35(5), 114–121.
38. Mishra, P., Godfrey, W. W., & Kumar, N. (2021). A green computing-based algorithm in software defined network with enhanced performance. In *2021 International conference on computing, communication, and intelligent systems (ICCCIS)* (pp. 953–958). IEEE.
39. Pokhrel, S. R. (2021). Software defined internet of vehicles for automation and orchestration. *IEEE Transactions on Intelligent Transportation Systems*, 22(6), 3890–3899.
40. Cui, X., Meng, Q., & Wang, W. (2020). A load balancing mechanism for 5G data centres. In *2020 International wireless communications and mobile computing (IWCMC)* (pp. 812–815). IEEE.
41. Nooruzzaman, M., & Fernando, X. (2021). Hyperscale data centre networks with transparent HyperX architecture. *IEEE Communications Magazine*, 59(6), 120–125.
42. <https://blog.leviton.com/new-switch-architectures-and-impact-40100g-migration-data-center>
43. <https://medium.com/@fiberstoreorenda/how-will-sdn-change-the-future-network-a0bbad6a3fld>

Artificial Intelligence Advancement for 6G Communication: A Visionary Approach



Javed Miya, Sandeep Raj, M. A. Ansari, Suresh Kumar, and Ranjit Kumar

Abstract Internet of the whole IoT primarily based totally clever issuer are gaining large recognition due to the ever-growing needs of wi-fi networks. This needs the appraisal of the wireless networks with progressed houses as next-generation communication systems. Regardless of the truth that 5G networks display extremely good potential to guide numerous IoE primarily based offerings, it isn't always right enough to satisfy the whole necessities of the brand-new smart programs. Furthermore, incorporating artificial intelligence in 6G will offer answers for extraordinarily complicated issues applicable to community optimization. Moreover, to add similarly fee to the destiny 6G networks, researchers are investigating new technologies, at the side of the and quantum communications. The requirements of destiny 6G wireless communications name for to help large records-driven applications and the developing style of customers. Furthermore, numerous destinies will hint to accomplish 6G-primarily based IoT networks are also highlighted.

Keywords 6G communication · Data science · Cognitive intelligence · Federated AI · Internet of everything · Wireless communications

J. Miya (✉) · S. Kumar · R. Kumar
Galgotias College of Engineering and Technology, Greater Noida, India
e-mail: javed.miya@galgotiacollege.edu; suresh.kumar@galgotiacollege.edu;
Ranjit.kumar@galgotiacollege.edu

S. Raj
Trinity Institute of Professional Studies, Dwarka, New Delhi, India
e-mail: sandeepbainy01@gmail.com

M. A. Ansari
GBU, Greater Noida, India
e-mail: ma.ansari@gbu.ac.in

1 Introduction

The view of our technical thought and the way we live now can be transformed by artificial intelligence, 6G connectivity and peace. In light of this, the market for communications technology in the future will curlicue between 2030 and 2040 [1, 2]. It will take decades to see the viable daybreak, as with many brainchild innovations [3]. The advantages and disadvantages of the new network technology have already been demonstrated by numerous studies. Full artificial intelligence will be integrated with 6G communiqué technology. Artificial intelligence will be a key component of the letter systems in the 6G communication network [4, 5]. Additionally, support for Extended Reality (XR) and Amplified Reality (AR) is anticipated [6]. The network system can be improved with the help of these technologies. In essence, Edge Computing and cloud-based architecture will be used for everything. The client only needs a quick internet connection; no servers, software, or hardware implementation are required. All network and flexibility will be acquired by cloud technology to deliver a real-time, data-driven environment. 6G will be able to provide amazing capacity and extremely low latency [7]. The world's communications sector is going through an astounding upheaval. It gains power with the use of a distributed network system. With its sprouting network system, cross-connection of a machine-type on a big scale, ultra-reliable, low-latency communications, and enhanced mobile broadband are all sought after goals [8]. Many pieces of modern hardware are also capable of handling significant data integration. Software-defined networking, assorted networks, and virtualization are new methods for cellular wireless networks to increase connection [9]. Control of the quick monitoring system with AI-powered network administration. Some tasks might be made simpler by using various cloud services, network virtualization, and other specialised equipment. However, over time, it causes more fragmentation and increases the need for tool regulation. We can create and change that sophisticated toolkit with the aid of 6G. The intricate amalgamation would be handled in a hierarchical sequence by edge AI over the administration system [5].

The current communication networking system is intelligently evolving to accommodate 6G communication. While 5G is the starting point for the next wave of communications technologies, Nayak & Patgiri [1, 2] illustrate the potential future applications of 6G in various fields, such as perception urban areas [10], intellectual vehicles, the Internet of Things, and smart cities. Researchers have focused on the 6G communication model even though 5G has yet to be widely implemented and gain experience [11]. In order to provide ubiquitous and dependable connectivity, additionally, the researchers have started combining the capabilities of AI with 6G communication networks. As a result, it is becoming into the foundation of society's digital transition [9, 12, 14]. By incorporating numerous technologies and offers, 6G communication technology makes everything connected. Additionally, it supports haptic, submersible, and holographic technologies. The Internet of Everything, including the Internet of Industrial Things, the Internet of Medical Things, the Internet of Nano-Things, and others, will be strengthened [7, 8, 13].

Consequently, with the aid of advanced AI, 6G letter technology can deliver on its promises.

2 Core Technologies of 6G Communication

Sixth-sense communication is the foundation of 6G communication. The technology will be three-dimensional, especially in terms of time, space, and frequency. A communication system powered by artificial intelligence will be 6G. The following characteristics are necessary for 6G communication technology: high data rate (about 1 Tbps), high waged recurrence (about 1 THz), low-slung start to finish latency (about 1 ms), steadfast high quality (10–9), high compactness (about 1000 km/h), and frequency of around 300 m [1, 2]. Additionally, enhanced augmented simulation and holographic communication will benefit intellectual network communiq  systems. The 3D forms of support will be provided by 6G with the aid of rising innovation, such as edge revolution, artificial acumen, distributed figuring, and blockchain [4]. There will be widespread integration of the 6G communication or generation. Through device to device, LEO, and satellite connectivity, 6G will provide further and more widespread integration [24]. For the communication organisation, 6G intends to combine computation, route, and detection. In the safety arena, 6G will handle refuge, confidentiality and shield of the enormous volume of data produced by billions of intellectual devices. “Intelligent gadgets” will replace the term “smart gadgets.” Fast connections to ultra-reliable low-latency communication are necessary for intelligent devices (URLLC). The fundamental requirements for 6G communication are 1 THz operational frequency, 1 Tbps data rate, 300 m frequency, and 1000 km flexibility range [1, 10]. The 6G architecture is 3D in terms of frequency, time, and space. End-to-end latency, radio-only delay, and processing delay are each 1 ms, 10 ns, and 10 ns, respectively, for 6G communication [5]. Based on artificial intelligence, 6G communication technology requires massively broad bandwidth machine type (mBBMT), broad mobile bandwidth and low latency (MBLL), and massive low latency machine type (mLLMT) [25]. Ten 6G communication trends are the main focus of Bi et al. [26]. An overview of artificial acumen in 6G communiq  is provided by Letaief et al. [27]. The primary themes discussed by Zhang et al. [28] are further-enhanced mobile broadband (FeMBB), awfully low-power communication, ultra-massive machine-type communication, long-distance and high-mobility infrastructures and highly dependable and low-latency communications. The below figure shows the feature possibility of 6G based applications (Fig. 1).



Fig. 1 The future possibility of 6G based applications

2.1 Eminence of Services

The eminence of service gauges how well a service performs overall in a given network area using 6G communication technologies. AI is at the heart of 6G communication's eminence of service. Excellent data speeds, ultra-reliable low-latency communication (URLLC), better portable wideband, ultra-enormous machine-type tradeoffs, long-distance and great mobility infrastructures and awfully low-power substitutions are all features of the 6G service that enable high QoS [2, 10]. Additionally, flexible, large transmission capacity and little idleness are part of quality of service [5, 17, 18, 19]. AI is largely responsible for the Quality of Service (QoS) in 6G communications, including AI-based physical layer security. As

organisational performance needs adjust to the expanding number of users, service quality is becoming more and more crucial. The most recent internet programmes and services demand enormous amounts of organisation execution and transmission capability, and their customers want them to provide higher consistency. With the enormous amount of equivalent force provided by 6G communication technology, this issue might be maintained. Associations must therefore send ideas and tactics that guarantee the best support. Quality of Service is also becoming more and more important as the Internet of Things develops. Machines are currently influencing businesses in the manufacturing sector to continuously notify customers of any impending problems [12]. Therefore, any delay in realising the problem could result in extremely costly mistakes. Data flows as swiftly as possible thanks to eminence of service, which enables the evidence stream to fulfil the needs of the organisation. Intelligent sensors are abundant in urban areas and are essential to the operation of the vast array of internet of things (IoT) undertaking structures.

2.2 The Standard of Experiences

A high eminence of service (EoS) and client-driven infrastructures with AI-assisted amenities are identified by the Quality of Experiences (QoE) framework (QoE). Since they need a high evidence rate with very little limpness, holographic infrastructures, augmented legitimacy, virtual realism, and quantifiable internet will be used to accomplish [6, 8]. In order to give users a higher Quality of Experiences (QoE), AI must combine with 6G communication. The quality of experience must also advance in sophisticated machines, smart devices, clever human services, intelligent automation, and many more areas. Only if 6G can fulfil all of its commitments will a great quality of life be achieved [12, 15]. The adoption of 6G will guarantee a high-quality experience, assisting the educational sector in sustaining a high-quality service experience. The degree of a patron's total level of fulfilment with aid, as seen from their perspective, is the nature of engagement. It aims to accept the client's abstract experiences with all of their complexities and human-subordinate elements, such as the material, transitory, transient, and financial ones. The way in which a particular case's Quality of Experience is determined depends less on the size or scope of the thing that needs to be measured and estimated [29]. Knowing which service parameters are essential to client satisfaction and assessing them from a standpoint as close to the client's perception as possible are the key issues. Therefore, in order to consistently enhance their services, professional organisations need to know how their clients feel about them. Figure 2 shows QoE monitoring for large traffic variations in business, public and residential areas in 6G network.

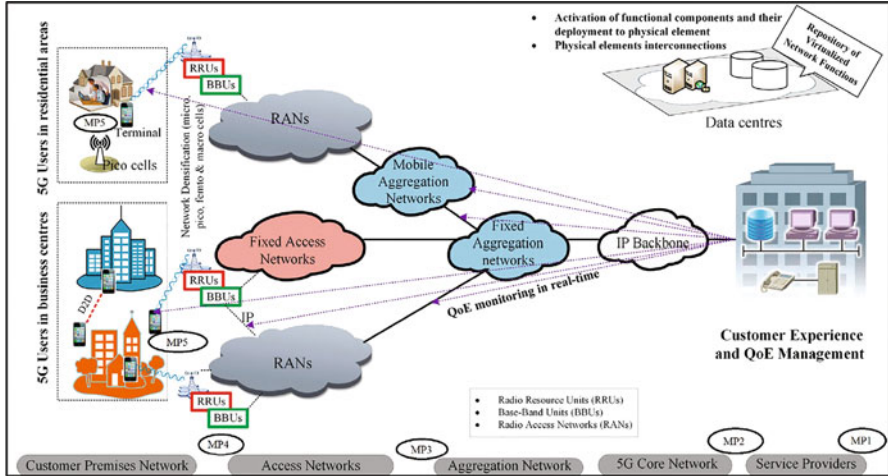


Fig. 2 QoE monitoring for large traffic variations in business, public and residential areas in 6G network

2.3 Quality of Lives

Quality of Experience (QoE) and Quality of Services (QoS) can be used to enhance Quality of Lives (QoL) (QoE). High quality of life will be made possible by 6G innovation via communication innovation [2]. When 6G innovation reaches all optimal boundaries, a high QoE can be achieved. A high evidence rate with very little sluggishness is necessary for holographic communications to deliver a high Eminence of Experience, extended authenticity, augmented authenticity and the quantifiable web [1]. Our social structures, organizations, and ways of life will alter significantly as a result of the 6G innovation, which will be a true simulation of intelligence-driven communication innovation. The 6G communication technology cannot deliver acceptable QoS, QoE, and Quality of Lives (QoL) without AI. 6G also claims to offer five senses of communication in order to create a high quality of experience. During the 6G era, all analogue devices will be converted to intelligent devices [4]. We will witness several advancements, from dazzling to cunning, with the appearance of synthetic intelligence and flexible communication. The Internet of Everything will take over from the Internet of Things after 2030, when it becomes intelligent and replaces it. The traditional mobile phone will be replaced by smartphones (as we go toward intelligent phones). Intelligent devices will be web-interactive, artificial intelligence-driven devices. Smart devices will therefore need to plan ahead, choose their course of action, and propose their cooperation with other smart devices [19]. To better the modern way of life, this help is essential to quality of life.

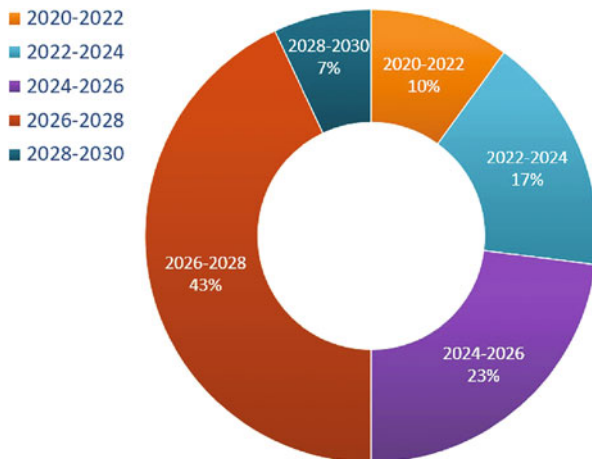
3 Internet of Everything

The Internet is become an integral aspect of our lives in this digital age. Capturing is the primary goal of high-level sensing. The gathered information is transformed into digital data, kept in a local cache, and sent in real-time to distant sites [4, 19]. On rare occasions, digital data may be further processed into signals and diffused to other hardware for dispensation [20]. By enabling us to gather all of our necessities in one place, the 6G network has the ability to usher in a communication revolution through the adoption of 6G communication technology. A broad idea called “the Internet of Everything” seeks to provide Internet of Things configurations a competitive edge. It empowers computerised change and presents decentralised frameworks objectively. It has been proposed that Internet of Everything (IoE) innovation will improve IoT business outcomes. The Internet of Things is now being advanced from many angles, including detecting efficacy, connecting devices, establishing communication channels, and gathering data from devices themselves. To employ the pure cycles and address the anticipated internet of things (IoT) challenges, these constraints are examined and removed. Internet of Everything (IoE) innovation hopes to provide a more comprehensive understanding of the same. The first stage of innovation is the Internet of Things, and the second is the Internet of Everything. They are linked to one another. Measure stack advanced’s main innovation is that it consistently adheres to the hyper-associated appropriated structure [30]. The Internet of Everything (IoE) technology’s main objective is to assist in transforming obtained statistics into substantial information-based abilities that can be promptly assimilated by the positioner’s internet of things applications. Applications for the Internet of Everything (IoE) include computerised sensor devices and interfaces for remote equipment, contemporary AI frameworks, as well as more sophisticated and constantly linked intelligent phones, and other types of distributed equipment that have recently become more automated and intelligent [11]. By having access to knowledge and extended systems administration possibilities, machines will frequently grow more intelligent. The below figure shows the prediction of worldwide internet usages from 2020–2030 for consecutive 2 years (Fig. 3).

4 Evolution from Smart to Intelligent

The Internet of Everything (IoE) willpower be perceptive and as a result, 6G will also be wise. As a result, all smart devices will evolve into intelligent devices, and these intelligent devices will be powered by AI [8, 13]. As a result, intelligent machines will be able to plan ahead, decide what to do, and offer to help out other machines [24]. Using 6G communication innovation and AI, there is a shift in perspective from brilliant to wise. We use multidimensional plan innovation to produce a very valuable product for our sector. To provide the finest response for the clients, it employs scientific and electronic approaches. Holographic user interface for the

Fig. 3 Prediction of worldwide internet usages from 2020–2030 for consecutive 2 years (GB/month)



6G remote systems management system is used to display the product. The use of remote multi-street availability is made for component and action communication. The simulated intelligence interface will allow for seamless communication over 6G for the human user interface. Human-Driven Administrations (HDA) require targets rather than inaccurate rate-dependability dormancy measurements [4]. The foundation Internet of Things can be used to connect dynamic structure components, and 6G communication technology will be used to deploy circulated characters. Brilliant sleuthing and implanted acquaintance would be employed to engross the model. The calculated direction would be taken by stem inquiry. For proper setting mindful structure, edge logic and multi-object Internet of Things are also used. A standard information assessment tool can predict how a business will perform in the future. Due to the 6G communication network’s high availability, massive amounts of data can also be controlled via holographic communication and virtual contact with another customer who may be a robot or human. For constant access to the material on the material web, rapid communication and ultra-reliable low-latency communication (URLLC) are necessary [22]. The cutting-edge 6G communication technology will be employed in conjunction with this development for distant operations. Experts will also benefit from it for inspection employing contact while they are not present. Three categories—work region, surface, and wearable—are used to categorise haptic Human-PC communication [16]. The inaccessible expert will have the option to operate or search utilising a virtual device in the working area. The device used to submit requests has a flat screen, such one on a tablet or compact. A request to interact with the patient can be given to the robot by moving the hand on the screen. For instance, haptic gloves are utilised by remote subject matter experts with wearable technology. Additionally assisting in providing human services amid a catastrophe is material progress [23]. Expanded Reality (AR) aids in integrating virtual and physical elements. Additionally, it is combined with a variety of material limitations, such as audio, visual, tactile, etc. Additionally, AR

offers constant association and displays 3D images of both virtual and actual objects [24]. When nothing is certifiable, augmented reality assists presenting a creative or virtual presence. To offer the extraordinary nature of the organisation, significant data rates are required.

5 AI Empowered 6G

Edge computing and artificial intelligence will be used in 6G communication technology to bring the server from the cloud closer to the users [21]. Both operationally and in terms of information and communications, the next generation of technology will change significantly from the one we currently use.

5.1 *Extreme Detail*

Thanks to a single network platform that is tailored to their requirements, customers can use their IP speeches and network types without changing the client data conscious attention. A Hyper-V host can host cybernetic machines that use the same IP outline thanks to the seclusion provided by the Gaps Network Virtualization (GNV) policy [12]. Web services will rapidly expand thanks to hyper-specific network technology. The escalation of web services faces a challenge. The rapid expansion of data sharing that has accompanied it has changed how we communicate with one another, the web, and our enterprises. Nearly all key developments in the previous 10 years have been driven by the espousal of web services and the underlying business and technological paradigms. But the rate at which these technologies are being adopted is not rectilinear. Although the first web amenities went live in the late 1990s, it took almost 10 years for them to become widely used. The proliferation of mobile devices, better-quality web content, and improved connectivity are all factors driving the rapid rise of web services and web-based applications. Web services currently play a large role in the digital economy, and as more advanced and potent applications are developed, we will see more of them in use. This has a significant advantage for hosting businesses. In addition to ensuring sure that they connect to their jobs using the same IP speech set as they do on their home-grown network, customers can construct strategy rules to isolate cybernetic machines from clients. Two benefits of offering a fully integrated and proven system are quick deployment times and an acceptable level of dependability [20]. This is due to the extensive number of tests that are performed on each component to guarantee that it functions well with every other component and results in an enterprise-level appliance [20]. It resembles a statistics centre in a container. The 6G announcement network will therefore have a significant impact on extreme details.

5.2 *Highly Capable*

The term “hyper-channel” technology was initially used to describe this tactic. It began as a response to faster data transfer technology replacing copper-based infrastructure. The phrase first appeared in the 1990, a time when grid speed was the main motivating factor. The phrase’s definition included the idea that a fast network could also carry more data than ever before. As a result of the emergence of highly-capable network technologies, some fundamental modifications to network protocols and network provisioning have been undertaken. The motivation behind these developments in the telecom and internet sectors is the need to manage network congestion in the twenty-first century. Terahertz spectrum is needed for the huge 6G expansion across all dimensions [16]. The virtual machine’s services are interrupted and present communication is broken by the change of address. Hyper Network Virtualization (HNV) now makes it easy to move virtual computers to other sub-networks on a regular basis [20]. The scene of an animate drifted cybernetic machine is rationalized and synchronised between hosts thanks to Hyper Network Virtualization’s continual communication with the migrated virtual machine. There have been no advancements made in the technology utilised for live virtual machine migration [21]. Virtual computers can be moved to different virtual sub-nets or locations without requiring a change to the network topology [21]. Hosting firms are easily able to shift client cybernetic machines across different data hubs without any downtime in the event of any conservation work at one of the compering sites, allowing clients to access cybernetic machines without any amenity interruption. This means that the capabilities of 6G communication technology will be substantially increased.

5.3 *Overly Sensitive*

Hyper sensing is a novel wireless network paradigm with many high-gain radio antennas on each node. With the available sensor technologies, the radio is intended for a single usage only. Thus, as compared to the human body, its power consumption is relatively high. The tuner sensors that are used to track anthropoid activity or vivacious cryptograms, on the other hand, are often built with lengthy life cycles in mind. Healthcare is the utmost common use for wireless sensors in general. Applications for healthcare may involve retrieving patient information or monitoring staff and patient health. A patient’s vital signs, such as heart rate, temperature, and blood pressure, are routinely measured using the beams that are recurrently inserted into the body. The sensors must last for many years in order to be used in various applications. With a high gain protuberance, the gain is maintained but the power consumption is far lower than with a traditional antenna. so that the sensor can receive more signals while using the same amount of transmitter power. Amplification of humans and robots is made simpler by 6G

connectivity. A hyper-converged architecture can be set up in a matter of weeks, maybe even a matter of days, as opposed to months for a standard solution that relies on part purchases. The interoperation of the components is given so much attention. Because the administration processes for the solutions are specifically designed for them, a lot of the complexity that typically makes it possible to oversee and ensure a datacenter is gone [21]. Hyperspectral line-scanning cameras use sensors to record reflected light across a slice of a picture [8]. Each spatial pixel, which contains the whole spectrum of information, is collected as motion occurs in one row each frame. A real force or an aerial deployment are the two methods for generating motion [12]. An additional choice is to deploy a hyperspectral beam in a immovable position to detect stationary objects exploiting pan-and-tilt or rotational segments [12]. As a result, the network's speed and adaptability are crucial for hyper sensing. The 6G network technology will have a significant impact on the large-scale industrial revolution [20]. The most cutting-edge networking system will experience a spin in the near future due to the technology's quick advancement.

6 Machine Learning with 6G

A growing number of people are using machine learning (ML) because of its many uses and capabilities. The characteristics of a system that is an explicit measured model are learned using computer algorithms referred to as machine learning models. These models can be used for tasks like classification, regression analysis, and interactions amongst intelligent agents in their environment. A qualified model is one that can pick up on a system's characteristics as it observes them [20]. Use some simple arithmetic calculations effectively to achieve the goal. Such advancement is made possible by the availability of trailblazing machine learning models, gigantic data sets, and strong computing supremacy [6, 21]. Installing ML requires a highly designed infrastructure with broad network liveness and real-time data dispensation. The major machine learning subtypes, such as supervised, unsupervised, and reinforcement erudition can all be alike integrated with 6G.

The use of machine learning (ML) benefits society. The last bit of metadata can be processed more efficiently and with less resources by using 6G. Machine learning has the ability to forecast certain limits and handle vast amounts of data. Machine learning has limitless potential. Each field has experienced significant benefits from machine learning that is compatible with 6G communication technology, which will soon do many activities [4]. Artificial intelligence and machine learning use a variety of methodologies, which has an impact on how we live our daily lives. In order to create a focused ecosystem, it may be important to ensure that cutting-edge technology like machine learning is applied correctly. Modern technologies are sought after by businesses in order to facilitate data analysis and improve decision-making. The vanguard of these activities is business intelligence (BI) software. The goalmouth of BI is to deliver the precise information to the precise people at the precise time. Autonomous operations are made possible by the software intelligence

Table 1 Details of the use cases for the comparative analysis of the 5G and 6G networks

Use case	5G	6G
Centre of gravity	User-centric	Service-centric
Augmented reality for industry in terms of peak rate and capacity	Low resolution and high-level tasks	High resolution with multi sensing and comprehensive level tasks
Tele-presence in terms of capacity	Limited scale and a high video quality	Mixed reality
Security surveillance, detection of defects in terms of positioning and sensing	External sensing with limited automation	Fully automated through the integrated radio sensing
Dynamic digital twins and virtual worlds	No	Yes
Data Centre wireless in terms of capacity and peak rate	No	Yes
Automation, distributed computing in terms of time synchronization	Micro second level tasks	High precision tasks at nano second level
Ultra-sensitive applications	Not feasible	Feasible
Zero energy devices	No	Yes
Groups of robots or drones in terms of low latency	Might be	Yes
Bio-sensors and AI	Limited	Yes
True AI	Absent	Present
Reliability	Not extreme	Extreme
Time buffer	Not real-time	Real-time
Satellite integration	No	Yes
Smart city components	Separate	Integrated

provided by 6G, which also continuously improves market results [20]. Table 1 shows the details of the use cases for the comparative analysis of the 5G and 6G networks [6, 112].

6.1 Supervised Learning

In supervised learning, models are trained using pre-existing data. Utilizing the previously available data source configuration and its predicted performance to calculate the quantity for the first half of the course stage [6, 9]. When the actual joint distribution of the input and output parameters is known and can be inferred from knowledge of the pertinent domain, supervised machine learning is most effective [6]. By employing the previously known set of inputs and their intended outputs, the coefficients of the downpour algorithms are taught through supervised learning. The optimum situation for supervised machine learning may be properly determined when the joint distribution of the input and output parameters

is accessible. Supervised learning at the physical layer can yield to optimal power distribution and a transient reduction in interference for transceiver communications [24]. Beyond the physical layer, shared networks, applications, transport, and other layer applications are also available for supervised learning. Consequently, the supervised learning process may be impacted by 6G.

6.2 Unsupervised and Semi-supervised Learning

Without knowing the names of the features, the archetypal learns how to bunch related data in unsupervised learning [16, 21]. In the 6G communiqué network, the system is trained using unsupervised learning even if there is no prior knowledge of the expected response from the system. Unsupervised learning is anticipated to be effective for a wide range of tasks, including point clustering, feature extraction, feature classification, dispersal estimation, and generation of dispersal-specific samples. In extremely complex vehicular infrastructures scenarios, less lucidity time limits the amount of time that may be spent in the physical layer [13]. The widespread adoption of 6G will make decision-making easier. Then, at the upper levels of the networking system, a plethora of unsupervised and semi-supervised learning approaches are useful for grouping, pairing, node clustering, or points for the best allocation of network resources. Access to annotated training data is restricted for semi-supervised learning. Real-time data relay over a high-frequency network like 6G is predominantly unlabeled, in contrast to unsupervised learning, which lacks access to annotated training data [6]. Although most of the knowledge is unlabeled, a small amount of annotated training data for semi-supervised learning is available. Model-based learning often optimises performance indices across accessible target functions with high computational efficiency. Channel equalisation and monitoring will be aided by semi-supervised learning [13]. Unsupervised learning will thus become more clever thanks to 6G.

6.3 Reinforcement Learning

During reinforcement learning, the operator connects the conditions with the situation and learns how to anticipate any repercussions on the action [22]. The sophistication and intelligence of the network have increased with the adoption of 6G telecommunications technology. Reward learning makes use of a variety of agents. To find the best programming settings and enhance network service quality, the agent can collaborate with each service station on the cellular network [9, 10, 31]. Ancillary control in this learning method, which may be characterised as a balance between supervised and unsupervised learning [20], is provided by the previous knowledge of the system's finest concert. The agent's long-term goal is to raise the cumulative reward. One of the numerous wireless obstacles that may

be phrased as reinforcement learning problems is resource allocation [20]. 6G would be used to build the networking infrastructure as a consequence. Innumerable deep reinforcement learning designs may be used to handle various difficulties on wireless networks.

The quantity and quality of the data being processed can be used to evaluate the machine learning algorithm. The paradigm for making decisions is directly impacted by networking constraints. Applications that have access to a substantial amount of historical training data can use batch-learning techniques [21]. These offline techniques, in which the data is manually gathered, labelled, and then analysed in batches, frequently have a limited amount of data at their disposal. To strengthen the 6G network, cutting-edge machine learning technology would be applied. The development of 6G connectivity will also heavily rely on quantum machine learning [11, 13]. Intellectual learning intellect and edge AI are used in quantum ML high accuracy real-time service is anticipated from quantum machine learning [10].

7 A Deep Learning 6G System

Every component of the deep learning scheme would be impacted by the 6G intellectual communiqué system. Deep Learning, a branch of machine learning used in artificial intellect, mostly uses unsupervised data. Unsupervised and supervised learning methods are combined in deep erudition. Automatic education features enable a system to cram complex functions that convert the input to the output from facts at various levels of construct without fully relying on hand-crafted topographies [19]. Deep learning has been researched for network planning and optimization, preventing intrusions, and diagnosing anomalies and faults [21]. Thanks to 6G networking technology, the system is able to gather and process data in real-time.

7.1 *Artificial Neural Network (ANN)*

A data dispensation structure inspired by biology called an artificial neural network is intended to learn new maneuvers from seen data [6]. One of the best deep learning methods is the artificial neural network (ANN). Numerous ways are able to utilise the enormous amount of data thanks to ANN, which has a neuron-like design. The brain level operation may be restricted again by the network's large computational capacity and connectivity [32]. The neural network is built from a number of layers [22]. Multi-layer perception also refers to the several underlying layers. In one layer, neurons are regarded as unimportant. A module for activation function is present in these nodes [12, 19]. The ANN neuron would compute complexity using cognitive intelligence. In an ANN, the way these interconnections are designed is crucial [13].

An ANN must be trained over a significant amount of data in order to extrapolate its processes and correctly handle any newfangled, unforeseen input statistics sample [22].

7.2 *Deep Neural Network (DNN)*

A unique mock neuron system proficient of classification and generalisation is the deep neural network. In this regard, deep learning can be viewed as a generalisation of the concepts of categorization and generalisation. We can think of deep erudition as a generalisation of human learning because human neurons have several layers [13]. Related to how the human brain can distinguish one image from another and recognise similar images, a deep neural network may also learn a model that can do the same. Just like the human brain, a vector serves as a synthetic sensory input for a deep neural network. Deep learning is sometimes referred to as a deep neural network since we supply this data as a matrix known as the input layer, also known as the hidden layer. Because a linear classifier can only learn linear categories, deep learning is able to learn enormously complex things like pictures, characters, and speech. The 6G contemporary communication system will be strengthened by deep networking. We can also rebuild neural networks using modern networking to expedite decision-making.

7.3 *Federated Instruction in the 6G*

A collaborative machine learning method called federated learning (FL) enables models to be disseminated among the participants in a learning activity without necessarily being aware of one another. Without explicitly sharing or transmitting the model parameters, it allows for the concurrent, cooperative training of machine learning models. Instead, depending on local data, each entity creates a local model and sends model strictures back to the alliance server for the final archetypal updates. The federated ensemble is then trained using these modifications. Separated and distributed among the entities are all the data and model parameters, which is conceptually comparable to federated learning. However, the primary distinction is that training is solely carried out locally as opposed to locally and centrally. Numerous attempts at amalgamated learning have been made in the past due to concerns about data concealment and the significant computational load of drill, but they have either not been adopted or have been unable to scale up. In order to make life simpler, 6G will be committed to self-learning and variety intelligent verdicts.

By reducing the amount of time needed and independently calming the condition of the training data, the amalgamated erudition technique has the advantage of speeding up the erudition progression. To make the network more cognitive, federated erudition and inferred erudition would be implemented. In federated

erudition, an algorithm is trained using a number of dispersed edge devices or servers that store local data samples without exchanging them. AI systems may learn from varied data sets kept across multiple sites thanks to a decentralised architecture.

Today's wireless communication networks are expected to see a fundamental paradigm shift towards intelligence from smartness and intelligence radio settings [12, 33]. The most imperative feature of deep learning occupations in these communiqué networks is not if they will be an important part of imminent networks, but rather when and how this enclosure can be triggered. Deep learning would be secondhand as a comprehensive strategy to improve the processing approaches for reckoning and decoding particular information on chronological blocks [34, 35]. Experience is the key to knowing how to recognise symptoms in critical situations, how to make the next decision, and what intervention to have. AI systems may learn from a large variety of data that is spread out across many locations thanks to federated learning. The approach enables numerous organisations to work together on model growth without needing to directly link important clinical data to one another [13]. By doing away with the requirement to pool data in a single location, federated learning decentralises deep learning. Instead, the model is trained across a number of iterations at numerous places. Federated learning also needs to be implemented carefully to keep patient data secure [19]. However, it can address some of the issues that approaches that allow for the combination of delicate epidemiological records face. In a federated erudition system, each of the multiple platforms that make up the erudition network has a copy of the archetypal stored on a supercomputer. The local data from the client is used by each unique device to train its copy of the archetypal. Then, a controlling computer receives the constraints and weights from each individual model, aggregates them, and updates the overall model. The training phase can then be repeated once more until the required level of uniformity is attained [36].

7.4 Black-Box

The black box process, commonly referred to as the cavernous erudition black box, is a kind of sophisticated machine learning technique in which researchers are not allowed entrée to the training data [6]. The great specificity of the grid where 6G communiqué technology can have an impact is wholly responsible for the black box's concealed restriction. Instead, then seeking to understand the inner workings of the learning algorithm, the training process' results are observed. In a real-world illustration, a black box is typically used on computers and/or smartphones to make judgments without the user being able to comprehend how it operates [32]. Since these forecasts are frequently applied in practical situations, the method is fundamentally uncertain because it is unaware of the surroundings. Dusky box machine erudition processes are useful for a number of tasks, including as forecasting an investor's future financial behaviour, spotting online fraud, or developing data mining systems. The black box process is most frequently used with

classifiers that have unidentified internal workings. A black-box classifier is given fresh training data after being trained on a set of data. A training set is the collection of training data used to build the model. The process would be more difficult with the help of the cutting-edge, mock intelligence-enabled 6G communiqué grid.

8 UAV with 6G

Unnamed UAVs (UAVs) are aircraft that may be remotely piloted and pre-programmed to do certain tasks. They can be thought of as miniature drones, balloons, intelligent drones, or aircraft. For proper military, surveillance, search and rescue, and other essential activities, cognitive intelligence networks are necessary, and 6G will be the technology that propels these developments. Identifying fires, accidents, traffic, etc. would be done by this kind of intelligent drone with the aim of continuously informing the police. Smart drones can also be used by the authorities to disperse crowds and spray gas [1]. With the use of artificial intelligence, many conventional systems, such as fire control, are anticipated to be replaced by unmanned aerial vehicles. This expertise will be used for 6G, which permits non-cellular infrastructures [7]. When the user tackle moves to begin with one cell before moving to the next, the call from the patron must now travel to the following cell [9]. Using mock intellect and a 6G connection, we can simply make drones niftier. Since they will be fortified with Drone-to-Drone and Drone-to-Infrastructures for communiqué, smart robots will be able to segment their familiarity and provide faster operational data transfer [1, 24].

Swarm drones, for instance, are employed in particular military activities. Examples of drone applications that can be used to track rim activity in immediate using ultra-high definition (uHD) cinematic broadcasts and 6G connectivity include Drone as Cop and Drone Shadowing on the rim of two Countries. All gadgets on a server can be connected through the most cutting-edge communication networks to enable alliance. The perfect 6G grid needs an intellectual cloud in order for mock intellect to be able to make decisions on its own. Drones are largely categorised by their elevation and type, so the increased communiqué capabilities of 6G expertise will be essential in this context. We also anticipate excellent living conditions and services. Additionally, Nayak and Patgiri found that linking 6G with closed-circuit video cameras (CCTV) showed some promise [1]. We are currently very safe thanks to the usage of CCTV for observation and security. CCTV is frequently employed as a court observer. CCTV is inefficient at both seeing the aforementioned risks and a wide variety of other things [2]. Lightweight shopping drone technology has decreased the cost of more contemporary aircraft, and the potential exponential evolution of these drones has increased the use of aerial photography and cameras, leading to the universality of automatically lifted frames. Avoid review and observation for business purposes.

Unmanned Aerial Vehicle, or UAV, is an acronym for an aircraft without a pilot. The UAV may operate autonomously in accordance with a pre-programmed

trip plan [37] or it may be a automatic jet that a pilot controls from a ground resistor station. Currently, drones are utilised for a variety of tasks, including attack and reconnaissance operations. Unmanned Aerial Vehicle (UAV) is a common shorthand used to describe the unmanned aerial vehicle system [29]. reflects the fact that in addition to the actual aircraft, these intricate systems call for ground stations and other parts. To more accurately represent the reality that these complex systems also include ground stations and other components in addition to the actual aircraft, the term “unmanned aerial vehicle” has been changed to “unmanned aircraft system” [29]. Drones have recently demonstrated potential as an expediently deployable in-flight wireless access platform for secure communications in a variety of brief social gatherings, military activities, and catastrophe circumstances. The most current drone technology developments allow wireless applications to create durable, transportable, and affordably priced drones [38]. Depending on their size and operational range, these unmanned aircraft systems are frequently divided into high-altitude platforms and low-altitude platforms (LAP). LAP is capable of operating up to several hundred metres in height. Throughout the procedure, the buzz is more likely to remain in a mid-range position as opposed to an extreme one. However, it is more likely that the drone’s motion will be homogeneous in its spatial phase. In order for drone technology to be completely functional, 6G will combine advanced network flexibility.

9 Automated Vehicle with 6G

With the aid of technology, we may now experience automatic vehicles, such as self-driving cars. They will improve fuel utilisation, course, and work competence because 6G communiqué innovation can provide unremitting services and intelligent cars are more conservative in their desire to forecast the imminent and financially handle problems [22]. The computer vision technique can be used to keep all the parts connected. In order to provide cutting-edge administration, 6G will look into dynamic range access and rely on AI to handle the heavy computation load. The automobiles will function intelligently and be able to communicate with one another for data sharing over 6G networking with a more interactive neural network link [11]. Self-sufficient vehicles (SSV) and connected automated vehicles are the two main types of robotic vehicles (RV).

9.1 Autonomous Automobiles

An self-sufficient vehicle is a vehicle that controls itself and keeps track of its surroundings exclusively through internal sensors. A machine that can move safely and comprehend its surroundings with almost no human input is called an autonomous motor. Autonomous vehicles have a variety of sensors built into them to

help them analyse their environment, including radar, sonar, GPS, and inertial extent units. Advanced rheostat systems will analyse sensory data to extricate between appropriate navigation courses, appropriate obstacles, and apposite messages [22]. The primary goal of functional expansion or improvement in the case of self-sufficient vehicles is to increase their safety [20]. There is no denying that autonomous vehicles are more secure than those operated by people. The remaining 10% of accidents and hazardous driving situations are caused by mechanical issues that are currently yonder our control [22]. The vast majority of car accidents are the consequence of transgression in some way. These wellbeing improvements will help variety human-driven vehicles safer before there are more self-driving cars on the road than human-driven ones.

9.2 Automated Allied Vehicles

A networked self-driving automobile is one that can identify adjacent vehicles with similar equipment. It was too conscious of the special characteristics of the nearby substructure, including as curves and intersections. One of the areas of automotive innovation that has received the greatest investigation is vehicle technologies [12]. The automotive technologies that are currently available are only a small portion of what is being produced for the future. There are currently a large number of linked vehicles on our roads, and a completely autonomous vehicle is one in which a driver is not necessary and is capable of transporting people. The technologies for connected automobiles, driverless vehicles, and progressive driver support systems overlap and contrast the technology, benefits, and difficulties of this developing industry [29]. With the use of connected car technology, vehicles may communicate with the local infrastructure. Most individuals are likely already familiar with connected vehicle technologies. To steer autonomous vehicles, several systems work together. The location of surrounding vehicles is tracked using radar sensors dispersed throughout the vehicle [21]. While keeping an eye out for pedestrians and other hazards, the camera recognises circulation lights, interprets circulation signs, and trails other vehicles. When parking, Light Range and Detection Sensors (LRDS) assist in locating autos and other vehicles. To control steering, acceleration, and braking, the vehicle's essential computer examines all the data from innumerable sensors. In addition to maintaining vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and a leading cloud, an autonomous linked vehicle system. With the use of this technology, driving distances can be covered in less time. The goal of 6G communiqué technology is to quickly connect the perfect network to the internet using the cloud to enable network connections. The network will be shrewd enough to cram from the past [37]. Deep learning, machine learning, and artificial neural networks can all be used to increase the efficiency and usefulness of machines. The 6G computing revolution will significantly strengthen analogue intelligence [12]. Manufacturers want to offer cars that will enhance their consumers' driving experiences while the automotive industry is on the cusp of

advancement. In the end, everything would change as a result of the formation and application of the linked automobile. From the jiffy the car door is opened to the moment it pulls up to the ultimate terminus, the allied aspects of the car would enable the motorist and passenger to have a significantly dissimilar experience, removing many of the current aching spots.

10 Data Science and 6G

An integral component of artificial intelligence is data science. It can analyse the dataset's insightful insider information to forecast future problems. The potent network capabilities of 6G [37] make it possible to optimise a whole data science lifestyle from beginning to end.

10.1 Descriptive Examination of Data

Descriptive analysis refers to the idea of applying analytical methods to explain or summarise a set of facts. One of the most common methods of data analysis, descriptive analysis is known for giving usable insights from previously unintelligible data. Unlike other types of data analysis, this descriptive study does not attempt to produce estimations. It does not manipulate the data in any manner to make it more pertinent; rather, it only draws conclusions from previous findings [23].

10.2 Data Analysis for Diagnostics

Evidence such as statistics breakdown, data sighting, data mining, collecting and causativeness are revealed through diagnostic data analysis. The growth of the market can be significantly altered by joining a smart grid network. Data exploration, drilldown, data mining, and correlations are frequently used in diagnostic analytics. In the discovery phase, analysts categorise the data sources that will aid in their interpretation of the results. Drilling down entails focusing on a specific widget or area of the details.

10.3 Analysing Prospective Data

The method of estimating the value of data in the future is known as prospective data analysis. Real-time and historical data will be combined with AI-enabled 6G technology to forecast the most important findings immediately. In a deep

network environment, we may develop accurate business intelligence tools and a complementing model with the use of artificial intelligence. Diagnostic analytics is one method through which we extract knowledge from our information and put it to use for us.

10.4 Predictive Data Analysis

As evidenced by the mobile phone, AI has effectively placed the entire world in our control. A well-established environment and the proper use of artificial intelligence can forecast data with more accuracy. Model optimization and refining would be a lot simpler with 6G communication network technologies. Predictive analytics is a term used to describe the routine of statistics and modelling to forecast yield based on available data and past performance [37]. Predictive analytic technologies will likely yield some findings with a high degree of accuracy. With the use of cutting-edge predictive analytics tools and models, any firm can now routine historical and real-time data to anticipate outlines and behaviours precisely in milliseconds, days, or years into the future. A type of data analytics called prognostic analytics uses past data and analytical methods like machine learning to forecast future outcomes [29]. The 6G communication technology's data-driven network architecture will put a particular emphasis on connectivity and data availability [11]. We may also infer how 6G and AI will motivate every marketing strategy to use its intelligent power more sparingly.

11 Artificial Robots with 6G

A revolution will soon be sparked by today's advanced artificial intelligence systems. To accurately execute jobs, people can work directly with machines. Industrial robots will get smarter as communication and artificial intelligence technologies advance [11]. For robots and sensors to operate accurately, matching must be incredibly strong and low-downtime, and swapping makes this process faster. To enable spontaneous replies, 6G edge nodes will manage complex calculations. In addition to supporting several robots and sensors, 6G features high-density communication capabilities [39]. We may claim that the 6G communication network will use embedded systems with quantum machine learning [4], wireless communication networks based on AI, and edge AI [10]. Although many of these limitations are still being researched, it is clear that 6G communiqué technology will become more widely used shortly. The field of robot automation can now consider more elements thanks to 6G connectivity expertise.

11.1 Robots with Expressive Intelligence

Learning the structure of the human brain is the biggest barrier to artificial intelligence. Deep neural grids are used to divide up vast expanses of data, like nerve cell, and keep calculating results in the future [38]. Robots aspire to take on more challenging tasks. Creating meaningful human interactions is one of these tasks, and it's also one of the hardest [10]. It will require a slice of networked, well-planned technology to fashion socially intelligent automata that can act as our friends, tutors, aides, and carers. Customized, potent, and lightweight software can greatly enhance a robot's capacity to interact with humans when it emanates to the user border. Many expensive and complex devices on the market failed because they were unintuitive to use. Any robot that interacts with humans directly should be programmed for a realistic, intuitive experience [10].

11.2 Industrial Robots

Artificial intelligence can be used to automate the entire industry with the aid of a connected intelligent network [10]. It has recently become possible because to the budding use of intelligent networks and smart gadgets. Additionally, the robots can communicate their data without involving any people. For continuing analysis and activity planning, it necessitates a smooth communication channel [11]. Artificial intelligence requires new robotics methods. To effectively use machine erudition and AI-guided jobs, the focus is on creating hardware and software hybrid systems that conduct operations with exceptional rapidity, dependability, sanctuary, and safety. Robots must have the highest level of precision and dependability, which requires AI. To determine the necessary timeframe for providing comprehensive robot maintenance, manufacturers in the robotics sector apply AI intelligence. Clients may do away with unnecessary breakdowns and the high maintenance costs that go along with them [10]. Robot efficiency is increased by carefully examining the information gathered by their sensors. These involve elements like electricity usage and transmission. The output of the AI algorithm can be used to instantly change the robot's programme. Robots with artificial intelligence (AI) are here to stay as manufacturing is prepared to enter a new industrial era. The development of automation and AI technologies will continue to evolve at a rapid rate, opening the door for wholly new and intriguing kinds of unresponsive technology, such as the self-sufficient, energetic machines of the future. The need for various technologies, such as mobile robots, surgical-robotic systems, and diagnostic robots, will rise dramatically as a result of the healthcare industry's rapid expansion. In 2025, it's predicted that the global healthcare market would be worth \$1,7 trillion. By 2025, it is projected that the logistics industry would expand to a value of over 1.2 trillion dollars. The booming e-commerce industry and the expansion of the retail sector are the main factors fuelling the growth of the logistics sector.

11.3 Robots in Healthcare

In order to escape the constraints of reality, The Internet of Intelligent Medical Things can be used (IIoMT). For example, experts can utilise remote assistance, pleasant appointments, or oral consultations to monitor remote surgery, and remote experts can use remote surgery to complete responsible tasks that call for prompt communication [10]. Hospital to Home (H2H), a mobile medical facility, will be built on an intellectual vehicle platform and rely on spare clinics, which will also have specialists and assistants on staff. This multipurpose medical facility will take the role of the recompenses of rescue vehicles, such as the steady identification of spare laptops in clinic accidents and arrival at the location. In order to test and validate procedures, intelligent wearable devices (IWDs) that are connected to the Internet can transmit mental and physical information [22]. Particularly for doctors and hospitals with access to massive data sets of potentially life-saving information, healthcare AI is applicable. This includes data on treatment strategies, their results, survival rates, and the speed of therapy gathered from millions of individuals, various geographic locations, and a wide range of often related health disorders. With the help of machine learning, new computational capacity may even create predictions to identify potential outcomes in addition to detecting and analysing patterns in both large and small data [10]. In hotels and shopping centres around the world, customer service robots are used. They can communicate with clients in a humane manner using artificial intelligence because of its ability to process natural language. Interestingly, prolonged interaction with people and customer service robots enhances their capacities. Open-source robotics systems with AI capabilities are made available for various industrial robots.

11.4 Smart Cities with Robotics

Services will be developed by artificial intelligence, which will also closely monitor crucial processes. Human robots will oversee logistical scanning thanks to the efficient management of the city's Internet connection [10, 40]. Other significant obstacles to the development of smart cities are comfort and security. Cognitive intelligence will also have a big impact on various businesses. According to research, Industry 4.0 has aided in the physical sector's transformation into a digital one, and 6G infrastructure's industrial automation will connect all different kinds of items, including mobile phones, tackles and robots [5]. It is projected that mock humans would steadiness and move in urban settings. Automation specialists are optimistic despite the widespread fears that people have about automata taking their jobs as significant as automata taking over the world. To combat the most challenging problems the world is currently facing, we will deploy robots. In order to understand climate change, we will analyse data from the masses, rain forests and atmosphere using algorithms. For societies that require it, we can arrange for

spare food, respond to natural tragedies and do widely more. Jobs like societal media researchers, data inventors, software inventors and mobile pushers were unimaginable a few years ago, but they currently exist. The deployment of a highly networked robotic system by Smart City Robotics is focused on the new cohort of Computerized Guided Vehicles (CGV) in order to realise this vision. AI robots are less expensive in the long term, however because to their complexity, the technology requires a substantial upfront investment. This technology will eventually become more affordable for enterprises as the demand for proprietary solutions decreases. Robots will be able to do difficult jobs with more accuracy and autonomy if more advanced AI systems are included into them. We may emphasise that a further adaptable 6G grid will drive our reduced toward an entirely computerized industry as a result. All of the tasks that currently require human labour can be replaced by robots. Robots will carry out crucial activities more precisely and with fewer human errors.

12 Security, Secrecy and Discretion

The highest level of security is typically an option for 6G transmissions. The aforementioned security concerns will be exacerbated by the addition of the Internet of Everything to 6G and the provision of additional management, including smart grids, residences, and emergency rooms [23]. Quantum communications will also make clear how crucial 6G security requirements are. The integrated AI will provide defence in 6G against hostile attacks. The physical layer security offered by 6G is noteworthy. Due of its higher terahertz (THz) [41], 6G may typically select the most important sanctuary level for transmission. The 6G security service is absolutely necessary for the mystery and security [29]. In any event, a crucial element of mystery, like the code word for financial balances, is needed when dealing with sensitive material. To solve the puzzle, use quantum cryptography. Protection is another major problem for those responsible for maintaining 6G [42]. The highest level of security assurance is particularly necessary for the human services sector [39]. New concepts, such as use cases, key performance metrics, models, factory procedures, and enabling innovation, current challenges, prospective fixes, openness, and exploration paths will become available with 6G [32]. Creating value by addressing human needs is 6G's primary objective. Low-latency grid connections, which enable quicker data allocation speeds, are necessary for smart grids. The 6G network will be more flexible and run on a customized platform.

Cellular networks will be optimised by 6G in terms of edges, hardware, and massive data transfer and dispensation. Each device can have a perspective that can be used either favourably or negatively with some quick machine learning study. If wicked devices possess this potential, resistance must be created on purpose. It can currently be applied in circumstances that are comparable. A security risk is posed by the safety of the several devices that are layered underneath [24].

13 Advanced AI/ML Techniques for 6G

Complicated optimization issues need be taken into account in 6G wireless systems under varying network topology and application circumstances. In these circumstances, we have two options for finding a solution. Utilizing an analytical optimization is the first strategy. In order to attain this, it may be obligatory to divide the original complex problem into a number of simpler problems or to relax the problem, It substitutes a simpler problem, such as a convex problem, for the original conundrum. However, because of this laxity and simplification, the outcome might be far from perfect. Utilizing AI/ML is the second strategy. Despite the fact that AI/ML techniques can solve the almost all of problems in 6G networks with complex topology and nonlinear components, careful AI/ML model design, selection of the proper training data set, and methods to safeguard and speed up the designed algorithm's convergence rate are still vital. As a result, we are unable to clearly separate them, and the choice of a solution strategy depends heavily on a particular scenario.

An in-depth discussion of AI-based 6G techniques can be found here. Also briefly covered was how AI/ML would be incorporated into the timeframe for 6G standardisation.

13.1 Reinforcement Erudition

Several Reinforcement Learning (RL) algorithms that use deep neuron networks and other methods to address RL difficulties, like exploration, have recently been suggested [45, 46]. A framework for RL that extends to multiagent systems called multiagent Reinforcement Learning (MARL) is also gaining popularity [47, 48]. From radio access control to network slicing, RL has been explored and viewed as a key technique for optimising several elements of 6G networks. This is due to the fact that proper modelling of 6G networks is expected to be challenging due to the complexity of the grid architecture and maneuver of such grids. Therefore, network optimization based on traditional model-based techniques will become less efficient. Because of this, Reinforcement Learning (RL) may offer a practical remedy for the effective functioning of such 6G networks. For example, network access issues [49–56], rate control issues [57, 58], resource management issues [59–62], and caching and offloading issues [63–73] have all been simplified using reinforcement learning. The authors of [49] devised dynamic multichannel access challenges as a partially observable RL scenario where a single agent selects one of N channels to carry out the broadcast leveraging just a portion of the channel. Additionally, by simulating several transmitters as various agents, authors in [50] increased each transmitter's contributions to the downlink spectral efficiency while reducing risks from other transmitters. A Q-learning-based transmission scheduling technique was put out in [74]. This technique increases system throughput by finding the most effective

way to transfer packets from different buffers. Deep reinforcement learning is used by [75] to enhance the dynamic adaptive streaming over HTTP experience. To surpass the most modern algorithms, the approach proposed in [75] incorporates feed-forward and recurrent deep neural networks. In addition to the examples given above, reinforcement learning has been used to enhance the performance of a number of optimization problems in the 6G network. In order to improve network efficacy in multichannel tuner networks, where each user is taught using DQN, multi-agent reinforcement learning was adopted in [76]. The method makes it possible for many users to train dispersedly in order to achieve the goal. As previously discussed, RL algorithms can be utilised to improve performance and adapt the network to match the needs of 6G.

13.2 Transfer Learning

The term “translational learning” (TL) describes the process of applying newly acquired knowledge to new activities. The core problem in many real-world situations, especially ML-based techniques, is data efficiency. By exploiting the learned information, TL can significantly enhance data efficiency.

TL has been studied and effectively used to streamline communication systems, particularly for ML-based optimization issues. As an illustration, [77] suggested a Reinforcement Learning framework for RAN energy-saving issues and integrated it with TL to hasten learning. For the target task, the policy that was trained for the source job was updated. Additionally, authors in [78] argued that by reducing the overall power consumption in fog RANs, resource management leveraging RL might be streamlined. To hasten learning, they also mixed the advised RL with TL. The Reinforcement Learning model parameters were moved to a archetypal that would be taught for milieus with different caching amenity capabilities after they had been learned on a source milieu with specific caching amenity capabilities. The authors of [79] forecasted the channel quality indicator over a number of wireless channels using the TL technique. They transfer the learned model by fixing the first layer of the previous archetypal and adding a new layer. As a result, TL can be utilised to improve energy efficiency, one of the requirements for the 6G Massive Radio Access Network (mRAN). A proposed model of mRAN intelligent system for 6G wireless communication is shown in Fig. 4. In addition to the aforementioned examples, TL has been extensively employed to enhance communiqué systems, such as resource management [80–84], channel guesstimate [79, 84, 85], caching [86–88] and localization [89–91] for boosting learning efficacy.

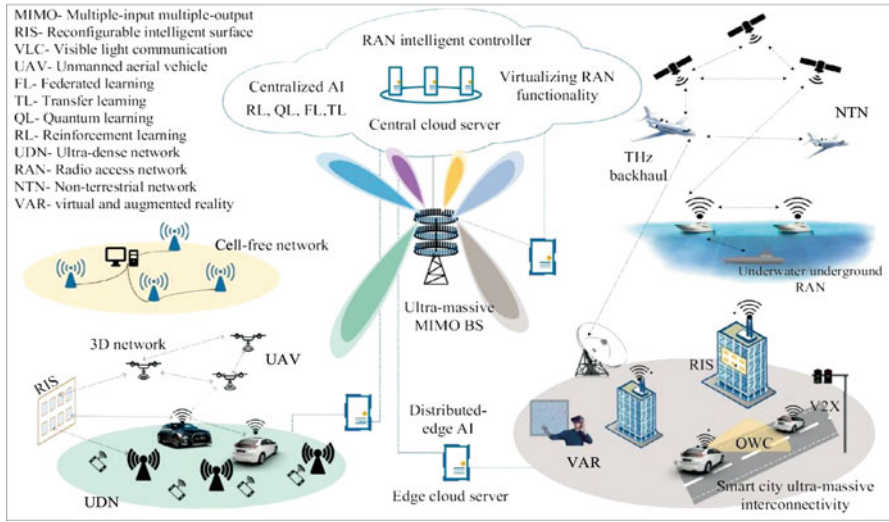


Fig. 4 A proposed model of mRAN intelligent system for 6G wireless communication

13.3 Federated Learning

Federated Learning (FL) does not share local data with other devices; instead, it learns ML models using data from dispersed devices. Federated Learning takes four crucial aspects into account, in contrast to conventional distributed optimization techniques [92]. First, the data distribution lacks IID. The FL design takes into account non-IID data distribution because actual data distributions from various devices vary depending on the local circumstances. The second issue is that the data are unequal, meaning that the amount of information gathered can differ depending on how each device is used. The third is the availability of widely used technology. The dispersed technology also has limited communication. Federated Learning assumes that the system has a set number of clients and a single central server based on the aforementioned features. The FL learns ML mockups using data from scattered devices rather than sharing local statistics with other diplomacies. Unlike other scattered optimization algorithms, FL deliberates four important factors [92]. First, there is no IID in the data distribution. The FL design takes into account non-IID data distribution because, in practise, data deliveries from different devices vary depending on the local surroundings. The second issue is that the data are unequal, meaning that the amount of information gathered can differ depending on how each device is used. The third is the availability of widely used technology. The dispersed technology also has limited communication. FL assumes that the system has a set number of clients and a single central server based on the aforementioned features. Additionally, in [93] authors proposed a FL-based content caching system that uses hierarchical architecture to prevent local private data leaks, and in [95]

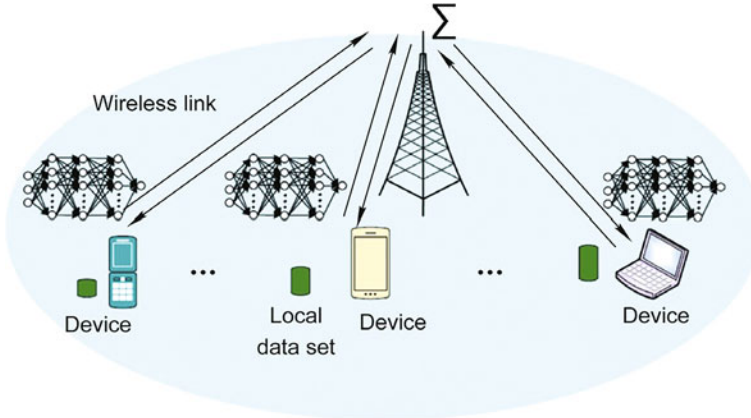


Fig. 5 Federation learning over a wireless communication network

authors introduced two-dimensional contract theory to allow interactions between the main system and mobile clients for resolving resource lopsidedness. The FL framework has embraced optimization concerns for MEC, such as edge securing and vehicular grid optimization, in addition to the aforementioned applications, to enable collaborative erudition of edge diplomacies. Authors in [94] presented a Federated Learning agenda to lower the power feasting of automotive users using dependability and low dormancy requirements. By using FL, severe events are reduced and the network-wide queues' tail distribution is properly educated. This component of FL makes it possible to meet 6G massive radio access network (mRAN) standards by increasing network adaptability. The below figure shows federation learning over a wireless communication network (Fig. 5).

13.4 Quantum Machine Learning

The more complicated applications envisioned for 6G, including ultra-large data volumes, pose a growing danger to security and privacy. Quantum computing-assisted communication uses the quantum key distribution (QKD) method to handle a variety of security issues and promise high dependability in 6G networks. Quantum key distribution (QKD), also graphene quantum cryptography, uses the quantum channel to emphasis was put a secret key between two authorised parties. ML integration and quantum cryptography are open to potentially difficult research paradigms. By delivering complete randomness over the 6G communication connections, quantum machine learning (QML) would improve cyber security. QML can help in a number of ways by improving sanctuary in terrestrial grids and NTN, ocean communiqué, THz, and optical communications [43]. In order to simplify the problem and boost performance, authors in [44] applied quantum

neural grids (QNG) and RL-aided QNNs (RL-QNN) to a user alliance problem in NOMA. When the average loss in this task was taken into account, RL-QNN produced better results. The quantum RL (QRL) can accelerate recital trade-offs such union, harmonizing, and optimality by harnessing the quantum superposition and parallelism features. Additionally, [96] has examined current developments in QRL approaches for spectrum assignment, management, and allocation.

13.5 Timeline for Integrating AI/ML into 6G Standards

A bridge between 5G and 6G technologies, 5G Innovative will be regulated in the 3GPP Rel-18 phase. 6G will be compatible with AI/ML technologies constructed on 5G Advanced. The AI/ML-Air Interface, on the other hand, will be applied to eliminate complexity and overhead or improve performance as one of the Rel-18 RAN1 elements. The NG RAN with AI support will also take care of indicating support and data collecting improvements. In order to meet imminent service requirements, the research piece for 6G, which will be released in Rel-20 segment (2025), needs to function better than 5G does. The regulation of AI/ML for 5G Progressive in Rel-18/19 will therefore help to loan the process of regulating AI/ML aimed at 6G.

14 Intelligent 6G Edge Technology

The end-users benefit from innovative and useful services provided by intelligent edge computing. Real-time service delivery, headlong data uploading and downloading, and intelligent responses to complex systems like factories and industries are a few of these. The task of uploading and processing the data in real-time is difficult at the edge, though. As a result, smarter solutions are required that might incorporate a hyperconnected edge. Figure 6 shows the application of edge computing technology.

14.1 Intelligent Hyper-connected Edge Networking

In this era of headlong networking and communiqué, edge calculation has shown promise in terms of data dispensation, stowing, and prompt delivery to culmination users. One of the biggest issues with cloud computing is how to move data over great distances for dispensation and storing [97]. Edge figuring, on the other hand, overcomes this difficulty by bringing computation close to the devices—such as computers, smartphones, medical equipment, and driverless vehicles. Edge computing strategies have been utilised in numerous telecommunications firms,

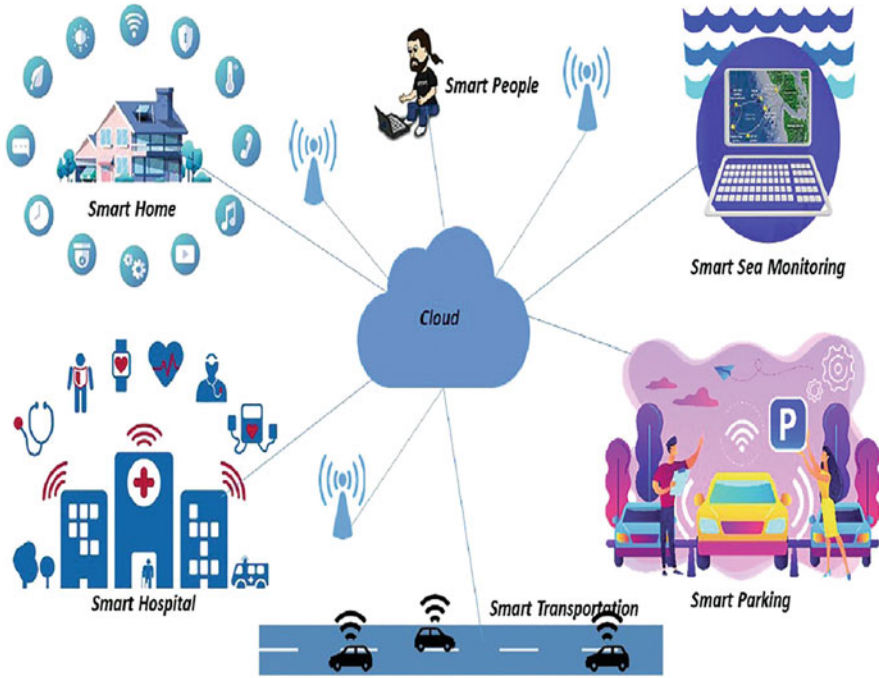


Fig. 6 Application of edge computing technology

the auto industry, smart factories, etc. because to these benefits. With the recently introduced 5G technology, instantaneous connectivity can be handled unfluctuating with a huge number of operators accessing the internet daily. A large number of edge devices must be deployed close to these users in order to provide services to them in real-time. Furthermore, the rise in internet users has made intelligent and hyperconnected edge networking necessary. The idea of transmitting all data to the edge is inefficient because devices like mobile phones, remote work sites, sensors, laptops, etc. produce a lot of data. The development of new lightweight approaches like FL and TL has lately addressed the challenge of training a machine learning algorithm locally and with fewer data samples. However, as these concepts were only codified, much more analysis and testing is required. Additionally, the concept of hyper-converged edge computing has been devised to lessen network stress by lowering the number of samples transmitted to the edge server. Thus, enhancing hyper-converged edge computing with intelligence can lower network load and enable the configuration of edge devices closer to the sources.

14.2 Split Computing with the Hyper-connected Edge Networks

Edge and mobile computing are connected through split computing. The application procedure is separated into head and tail parts, with the head half rolling on mobile devices and the tail half launching on the edge. To maximise performance, the authors of [98] divided their artificial neural network model across the subnets. Edge processing stint, device processing time, and connection delay are the three factors that make up a task's overall completion time in split computing. The fundamental challenge of split computing is lowering communication delay while leaving only a small amount of computation on the device to transmit data to the server and reduce dispensation time to that of superiority and mobile computation. Split computation can, in some unusual circumstances, fall short of the standards set forth in [99]. The fundamental difficulty is allocating a portion of the total task to a device with a low processing capability, which results in the transmission of a negligible amount of information. Deep neural networks are currently used in the picture classification sector where split computing has been implemented. This results in a number of issues, including compression-related issues, because the board entity is an input. Additionally, split computing has advantages in a lot of other unknown sectors. Real-time nursing of smart healthiness care systems with abiliment sensors is one efficient split computing application, which makes use of devices like home routers and cell phones as sensors.

14.3 Edge-Offloading Technology

The two components of wireless edge-offloading technology are network traffic offloading and computation offloading. An effective technique for proactively caching popular content was proposed by Femtocaching, which also provided a revolutionary wireless edge caching architecture. The architecture features several edge caches placed at BSs that update well-liked material. The femtocaching architecture was expanded in a number of studies [100, 101]. Multiple BSs worked together to augment the location of the material in the cooperative edge caching system created by Lyu et al. In a graded cloud-edge storing paradigm, the haze and superiority store files in accordance with local and global file popularities, respectively. Sadeghi et al. [101] took this into consideration. A Markov chain model was used to model the spatiotemporal popularity changes, and RL was used to solve them. Edge networks, which are comparable to cloud computing and have lower end-to-end latency than central clouds, offload computational chores for end devices. Kwak et al. [102] have considered a mobile code offloading architecture with a variety of job kinds that saves energy in end devices. Then, using a game-theoretical method, Chen et al. [103] investigated the code offloading choices made by several mobile users who share a wireless connection. In both competitive and

cooperative scenarios between the offloading service provider and mobile clientele, Kim et al. [104] created an integrated edge-offloading architecture by leveraging the Lyapunov drift-plus-penalty approach. The main concept for addressing dynamic changes in mobile traffic needs and computations for future 6G networks is learning-based resource allocation. Additionally, it's crucial to research where to place edge devices and how much money to spend on them.

15 Conceptualization of the Intelligent PHY Layer for 6G

It is crucial to address the issue of narrow spectrum resources in edge computing, future multiple access and new opportunistic spectrum technologies, new channel coding technologies, and propose highly accurate intelligent channel prognostication technologies in order to meet the higher peak rate, capacity requirements, and spectrum efficiency requirements of 6G satellite phones. This section contains further details on a variety of topics.

15.1 Management of Independent Radio Resources Based on Cognitive Intelligence

In order to transition from the centralized approach of the 5G RAN to an edge computing structure for the 6G RAN, it is necessary to think about effective strategies to manage the scarce radio resources. One such strategy is learning, justification, and optimization engines in cognitive intelligent-based autonomous radio resource management. It is challenging to quickly allocate resources within a constrained timescale, and the optimization process for interference and cohabitation between cell membranes loses essential information about the ideal solution [105]. Data mining and interference analysis based on the Monte Carlo technique are therefore used to improve the performance of perceptual intelligent-based autonomous managing radio resources [106].

15.2 Intelligent Modulation and Coding of the Channels

Widespread interest has been shown in canal coding techniques grounded on polar, low-density parity-check and turbo encryptions. In addition to being infinitely close to the Shannon limit, they have also made enormous strides in error correction, code proportion and code extent reconstruction, supported amalgam automatic re-transmission appeals and complexity [107]. Because they can be easily supported by hybrid automatic repeat requests and are flexible in terms of inscribing

rate, inscribing duration, and decipherment delay, LDPC cryptographs have been used to safeguard 5G canons [108]. Recently, data-driven scrutiny, corollary, and supervisory technologies have been applied in communication system performance improvement employing AI technology [109]. AI-driven conduit coding mechanisms use AI-based outlines, DL, NN, and evolutionary procedures, whereas conventional channel coding mechanisms primarily rely on coding theory to increase coding performance.

15.3 Channel Estimate with AI

With reliable channel municipal information for the effective claim of MIMO, 6G will espouse the ultra-large-scale MIMO expertise to satisfy the demands of larger communication capacity. The ultra-large-scale protuberance array's channel estimation issue is more challenging due to the array's multiple antennas, nevertheless. The primary proposed channel estimation techniques are distributed compressed sensing-based, sparse analysis-based approximation message forwarding, and support detection-based [110]. AI technology can be used to boost channel estimation performance, and a neural network can be used to build AI-based channel estimation technology. Before equally dividing and adding the data bits to the neural network, the input signal is split into its pilot and data components. The channel characteristics may be ascertained utilising the signal's input-output relationship after producing the corresponding data sequence.

15.4 Intelligent Spectrum Sharing and Multiple Access

The sharing of bandwidth resources among mobile users is governed by multiple access technology. The frequency ranges 3–6 GHz and 24–50 GHz are now being used by 5G mobile communication technologies. It is crucial to make the choice to be more compatible with future multiple access technologies and the use of new spectrum sharing technologies given the increasing peak rate, capacity needs, and spectrum efficiency requirements of 6G satellite phones. By extending the accessible spectrum, such as the THz and visible light spectra, the requirement for using spectrum resources in the next 6G system must be addressed. The spectrum sensing rules must also be changed, and the permissible carrier usage status must be broken. To harness the benefits of spectrum resources, it is also necessary to maintain the more flexible allocation and use of spectrum. In the forthcoming 6G era, spectrum sharing could be crucial. Sharing unlicensed spectrum in the mmWave bands, especially above 60 GHz, is viable to meet the high-performance requirements of the 6G ultra-large-scale internet of things [111].

16 Conclusion

We can infer from the agreement that 6G communications will, in the near future, be combined to develop an intelligent communication system using artificial intelligence that will meet the needs of everyday people. Every facet of the intelligent network, including mock general astuteness, tailored mock intelligence, and mock superintelligence, will be influenced by the cerebral model of the grid architecture. The intelligent network will be entirely AI-driven. More crucially, the numerous potential supporting technologies will guarantee high levels of service quality and user experience to transform society into an AI-driven smart city. As a result, we may assume that in the near future, both AI and 6G will become more effective and dependable for end users. An entirely new age in digital communication technology will be ushered in by the convergence of AI and 6G networks.

A new assembly paradigm of trade internet-of-things, internet-of-people, and cyberspace global economic amenities has emerged as a result of the recent evolution of the internet-of-everything. Future wireless networks will see a huge increase in mobile data traffic as a result of the services' various and multifaceted demand for nearly instantaneous, universal, and unlimited access. So, in this study, we looked into the key requirements supporting emerging technologies, networking solutions at the superiority and physical layer, and the driving concerns related to the intelligent 6G massive radio access network (mRAN) architecture. We also gave network intelligence priority and conducted a thorough examination of the possible applications of AI and ML mockups in network management, store allocation, gamut sharing, superiority schmoozing, and safekeeping. We also made clear the most important research problems and obstacles in each 6G massive radio access network (mRANs) related area.

References

1. Nayak, S., & Patgiri, R. (2020). 6G communication: Envisioning the key issues and challenges, EAI endorsed trans. *Internet Things*, 6(24), 166959. <https://doi.org/10.4108/eai.11-11-2020.166959>
2. Nayak, S., & Patgiri, R. (2020). 6G communications: A vision on the potential applications, *arXiv:2005.07531*.
3. Bin Ahammed, T., & Patgiri, R. (2020). 6G and AI: The emergence of future forefront technology. In *2020 Advanced communication technologies and signal processing* (pp. 1–6). ACTS. <https://doi.org/10.1109/ACTS49415.2020.9350396>
4. McMahan, B., Moore, E., Ramage, D., Hampson, S., & yArcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In A. Singh & J. Zhu (Eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, in: *Proceedings of Machine Learning Research* (Vol. 54, pp. 1273–1282). PMLR.
5. Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y., & Kim, D. I. (2019). Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys and Tutorials*, 21(4), 3133–3174.

6. Viswanathan, H., & Mogensen, P. E. (2020). Communications in the 6G era. *IEEE Access*, 8, 57063–57074.
7. Ali, S., Saad, W., Rajatheva, N., Chang, K., Steinbach, D., Sliwa, B., Wietfeld, C., Mei, K., Shiri, H., Zepernick, H. -J., Chu, T. M. C., Ahmad, I., Huusko, J., Suutala, J., Bhadauria, S., Bhatia, V., Mitra, R., Amuru, S., Abbas, R., Shao, B., Capobianco, M., Yu, G., Claes, M., Karvonen, T., Chen, M., Girnyk, M., & Malik, H. (2020). 6G white paper on machine learning in wireless communication networks. *arXiv:2004.13875*.
8. Kasgari, A. T. Z., Saad, W., Mozaffari, M., & Poor, H.V. (2019). Experienced deep reinforcement learning with generative adversarial networks (GANs) for model-free ultra reliable low latency communication, *arXiv: 1911.03264*.
9. Sharma, P., Liu, H., Wang, H., & Zhang, S. (2017). Securing wireless communications of connected vehicles with artificial intelligence. In *2017 IEEE international symposium on technologies for homeland security, HST* (pp. 1–7). IEEE.
10. Nayak, S., & Patgiri, R. (2021). 6G communication technology: A vision on intelligent healthcare. In *Health Informatics: A Computational Perspective in Healthcare* (pp. 1–18., ISBN: 978-981-15-9734-3). Springer. https://doi.org/10.1007/978-981-15-9735-0_1
11. Piran, M. J., & Suh, D. Y. (2019). Learning-driven wireless communications, towards 6G, *arXiv:1908.07335*.
12. Nawaz, S. J., Sharma, S. K., Wyne, S., Patwary, M. N., & Asaduzzaman, M. (2019). Quantum machine learning for 6G communication networks: State-of-the-art and vision for the future. *IEEE Access*, 7, 46317–46350.
13. Sun, Y., Liu, J., Wang, J., Cao, Y., & Kato, N. (2020). When machine learning meets privacy in 6G: A survey. *IEEE Communications Surveys and Tutorials*, 22(4), 2694–2724.
14. Yang, H., Alphones, A., Xiong, Z., Niyato, D., Zhao, J., & Wu, K. (2020). Artificialintelligence-enabled intelligent 6G networks. *IEEE Network*, 34(6), 272–280.
15. Zhou, Y., Liu, L., Wang, L., Hui, N., Cui, X., Wu, J., Peng, Y., Qi, Y., & Xing, C. (2020). Service aware 6G: an intelligent and open network based on convergence of communication, computing and caching. *Digital Communications and Networks*, 6(3), 253–260.
16. Simeone, O. (2018). A very brief introduction to machine learning with applications to communication systems. *IEEE Transactions on Cognitive Communications and Networking*, 4(4), 648–664.
17. Chen, Y., Liu, W., Niu, Z., Feng, Z., Hu, Q., & Jiang, T. (2020). Pervasive intelligent endogenous 6G wireless systems: Prospects, theories and key technologies. *Digital Communications and Networks*, 6(3), 312–320.
18. Wang, M., Zhu, T., Zhang, T., Zhang, J., Yu, S., & Zhou, W. (2020). Security and privacy in 6G networks: New areas and new challenges. *Digital Communications and Networks*, 6(3), 281–291.
19. Chen, M., Yang, Z., Saad, W., Yin, C., Poor, H. V., & Cui, S. (2019). A joint learning and communications framework for federated learning over wireless networks, *arXiv:1909.07972*.
20. Wang, J., Jiang, C., Zhang, H., Ren, Y., Chen, K., & Hanzo, L. (2020). Thirty years of machine learning: The road to Pareto-optimal wireless networks. *IEEE Communications Surveys & Tutorials*, 22(3), 1472–1514.
21. Liu, Y., Bi, S., Shi, Z., & Hanzo, L. (2020). When machine learning meets big data: A wireless communication perspective. *IEEE Vehicular Technology Magazine*, 15(1), 63–72.
22. Letaief, K. B., Chen, W., Shi, Y., Zhang, J., & Zhang, Y. A. (2019). The roadmap to 6G: AI empowered wireless networks. *IEEE Communications Magazine*, 57(8), 84–90.
23. Saad, W., Bennis, M., & Chen, M. (2020). A vision of 6G wireless systems: Applications, trends, technologies, and open research problems. *IEEE Network*, 34(3), 134–142.
24. Zappone, A., Di Renzo, M., & Debbah, M. (2019). Wireless networks design in the era of deep learning: Model-based, AI-based, or both? *IEEE Transactions on Communications*, 67(10), 7331–7376.
25. Gui, G., Liu, M., Tang, F., Kato, N., & Adachi, F. (2020). 6G: Opening new horizons for integration of comfort, security and intelligence. *IEEE Wireless Communications*, 27, 1–7. <https://doi.org/10.1109/MWC.001.1900516>. ISSN: 1558-0687.

26. Bi, Q. (2019). Ten trends in the cellular industry and an outlook on 6G. *IEEE Communications Magazine*. ISSN: 1558-1896, 57(12), 31–36. <https://doi.org/10.1109/MCOM.001.1900315>
27. Letaief, K. B., Chen, W., Shi, Y., Zhang, J., & Zhang, Y. A. (2019). The roadmap to 6G: AI empowered wireless networks. *IEEE Communications Magazine*. ISSN: 1558-1896, 57(8), 84–90. <https://doi.org/10.1109/MCOM.2019.1900271>
28. Zhang, Z., Xiao, Y., Ma, Z., Xiao, M., Ding, Z., Lei, X., Karagiannidis, G. K., & Fan, P. (2019). 6G wireless networks: Vision, requirements, architecture, and key technologies. *IEEE Vehicular Technology Magazine*. ISSN: 1556-6080, 14(3), 28–41. <https://doi.org/10.1109/MVT.2019.2921208>
29. Alsharif, M. H., Kelechi, A. H., Albreem, M. A., Chaudhry, S. A., Zia, M. S., & Kim, S. (2020). Sixth generation (6G) wireless networks: Vision, research activities, challenges and potential solutions. *Symmetry*, 12(4), 676.
30. Sharma, P. K., Deepthi, D., & Kim, D. I. (2019). Outage probability of 3-D mobile UAV relaying for hybrid satellite-terrestrial networks. *IEEE Communications Letters*, 24(2), 418–422.
31. Jung, M., & Saad, W. (2021). Meta-learning for 6G communication networks with reconfigurable intelligent surfaces. In *ICASSP 2021–2021 IEEE international conference on acoustics, speech and signal processing, ICASSP* (pp. 8082–8086). IEEE. <https://doi.org/10.1109/ICASSP39728.2021.9413598>
32. Gui, G., Liu, M., Tang, F., Kato, N., & Adachi, F. (2020). 6G: Opening new horizons for integration of comfort, security and intelligence. *IEEE Wireless Communications*, 27, 1–7.
33. Liu, Y., Yuan, X., Xiong, Z., Kang, J., Wang, X., & Niyato, D. (2020). Federated learning for 6G communications: Challenges, methods, and future directions. *China Communications*, 17(9), 105–118. <https://doi.org/10.23919/JCC.2020.09.009>
34. Adeogun, R., Berardinelli, G., Mogensen, P. E., Rodriguez, I., & Razzaghpour, M. (2020). Towards 6G in-X subnetworks with sub-millisecond communication cycles and extreme reliability. *IEEE Access*, 8, 110172–110188. <https://doi.org/10.1109/ACCESS.2020.3001625>
35. Patgiri, R., & Ahmed, A. (2016). Big data: The v's of the game changer paradigm. In *2016 IEEE 18th international conference on high performance computing and communications; IEEE 14th international conference on smart city; IEEE 2nd international conference on data science and systems (HPCC/SmartCity/DSS)* (pp. 17–24). <https://doi.org/10.1109/HPCC-SmartCity-DSS.2016.0014>
36. Sergiou, C., Lestas, M., Antoniou, P., Liaskos, C., & Pitsillides, A. (2020). Complex systems: A communication networks perspective towards 6G. *IEEE Access*, 8, 89007–89030. <https://doi.org/10.1109/ACCESS.2020.2993527>
37. Elmeadawy, S., & Shubair, R. M. (2019). 6G wireless communications: Future technologies and research challenges. In *2019 International conference on electrical and computing technologies and applications, ICECTA* (pp. 1–5). IEEE.
38. Sharma, P. K., & Kim, D. I. (2020). Secure 3D mobile UAV relaying for hybrid satellite-terrestrial networks. *IEEE Transactions on Wireless Communications*, 19(4), 2770–2784.
39. Calvanese Strinati, E., Barbarossa, S., Gonzalez-Jimenez, J. L., Ktenas, D., Cassiau, N., Maret, L., & Dehos, C. (2019). 6G: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication. *IEEE Vehicular Technology Magazine*, 14(3), 42–50.
40. Jamil, S. U., Arif Khan, M., & Rehman, S. U. (2020). Intelligent task off-loading and resource allocation for 6G smart city environment. In *2020 IEEE 45th conference on local computer networks, LCN* (pp. 441–444). IEEE. <https://doi.org/10.1109/LCN48667.2020.9314819>
41. Attanasio, B., La Corte, A., & Scatà, M. (2021). Evolutionary dynamics of MEC's organization in a 6G scenario through EGT and temporal multiplex social network. *ICT Express*, 7(2), 138–142.
42. Yeh, C., Do Jo, G., Ko, Y.-J., & Chung, H. K. (2022). *Perspectives on 6G wireless communications*. ICT Express.
43. Nawaz, S. J., Sharma, S. K., Wyne, S., Patwary, M. N., & Asaduzzaman, M. (2019). Quantum machine learning for 6G comm. netw.: State-of-the-art and vision for the future. *IEEE Access*, 7, 46317–46350.

44. Narottama, B., & Shin, S. Y. (2021). Quantum neural networks for resource allocation in wireless communications. *IEEE Transactions on Wireless Communications*, 21(2), 1103–1116.
45. Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *2018 International conference on machine learning, ICML* (pp. 1–10). PMLR.
46. Han, S., & Sung, Y. Diversity actor-critic: Sample-aware entropy regularization for sample-efficient exploration. In *2021 International conference on machine learning, ICML* (pp. 1–12). PMLR.
47. Foerster, J. N., Assael, Y. M., Freitas, N. D., & Whiteson, S. (2016). Learning to communicate with deep multi-agent reinforcement learning. In *2016 30th Conference on neural information processing systems, NIPS* (pp. 1–9). PMLR.
48. Lee, H., Kim, E., Kim, H., Na, J. H., & Choi, H.-H. (2021). Multi-agent Q-learning based cell breathing considering SBS collaboration for maximizing energy efficiency in B5G heterogeneous networks. *ICT Express*, 8(4), 525–529. <https://doi.org/10.1016/j.ict.2021.09.006>
49. Wang, S., Liu, H., Gomes, P. H., & Krishnamachari, B. (2018). Deep reinforcement learning for dynamic multichannel access in wireless networks. *IEEE Transactions on Cognitive Communications and Networking*, 4(2), 257–265.
50. Nasir, Y. S., & Guo, D. (2019). Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks. *IEEE Journal on Selected Areas in Communications*, 37(10), 2239–2250.
51. Man, C., et al. (2018). Reinforcement learning-based multiaccess control and battery prediction with energy harvesting in IoT systems. *IEEE Internet of Things Journal*, 6(2), 2009–2020.
52. Hao, Y., et al. (2018). Deep reinforcement learning for resource allocation in V2V communications. In *2018 IEEE International Conference on Communications, ICC*. IEEE.
53. Chen, M., Saad, W., & Yin, C. (2017). Liquid state machine learning for resource allocation in a network of cache-enabled LTE-U UAVs. In *2017 IEEE Global Communications Conference* (pp. 1–6). IEEE.
54. Challita, U., et al. (2018). Proactive resource management for LTE in unlicensed spectrum: A deep learning perspective. *IEEE Transactions on Wireless Communications*, 17(7), 4674–4689.
55. Liu, S., Hu, X., & Wang, W. (2018). Deep reinforcement learning based dynamic channel allocation algorithm in multibeam satellite systems. *IEEE Access*, 6, 15733–15742.
56. Nan, Z., et al. (2018). Deep reinforcement learning for user association and resource allocation in heterogeneous networks. In *2018 IEEE Global Communications Conference, GLOBECOM*. IEEE.
57. Mao, H., Netravali, R., & Alizadeh, M. (2017). Neural adaptive video streaming with pensieve. In *Proceeding of the conference of the ACM special interest group on data communication* (pp. 197–210). Association for Computing Machinery.
58. Ferreira, P. V. R., et al. (2018). Multiobjective reinforcement learning for cognitive satellite communications using deep neural network ensembles. *IEEE Journal on Selected Areas in Communications*, 36(5), 1030–1041.
59. Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2016). Resource management with deep reinforcement learning. In *15th ACM Workshop on Hot Topics in Networks* (pp. 1–7). Association for Computing Machinery.
60. Peng, H., & Shen, X. (2021). Multi-agent reinforcement learning based resource management in MEC- and UAV-assisted vehicular networks. *IEEE Journal on Selected Areas in Communications*, 39(1), 131–141.
61. Xu, Z., Wang, Y., Tang, J., Wang, J., & Gursoy, M. C. (2017). A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs. In *2017 IEEE International Conference on Communications, ICC* (pp. 1–6). IEEE.
62. Li, T., et al. (2018). Model-free control for distributed stream data processing using deep reinforcement learning. *Proceedings of the VLDB Endowment*, 11(6), 705–718.

63. He, Y., Yu, F. R., Zhao, N., Leung, V. C. M., & Yin, H. (2017). Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach. *IEEE Communications Magazine*, 55(12), 31–37.
64. He, Y., et al. (2017). Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach. *IEEE Communications Magazine*, 55(12), 31–37.
65. Zhong, C., et al. (2018). A deep reinforcement learning-based framework for content caching. In *52nd Annual conference on information sciences and systems, CISS* (pp. 1–6). IEEE.
66. He, Y., et al. (2017). A big data deep reinforcement learning approach to next generation green wireless networks. In *2017 IEEE global communications conference* (pp. 1–6). IEEE.
67. He, Y., et al. (2017). Optimization of cache-enabled opportunistic interference alignment wireless networks: A big data deep reinforcement learning approach. In *IEEE international conference on communications, ICC* (pp. 1–6). IEEE.
68. He, X., et al. (2018). Green resource allocation based on deep reinforcement learning in content-centric IoT. *IEEE Transactions on Emerging Topics in Computing*, 8(3), 781–796.
69. He, Y., Zhao, N., & Yin, H. (2017). Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach. *IEEE Transactions on Vehicular Technology*, 67(1), 44–55.
70. Zhang, C., et al. (2018). A deep reinforcement learning based approach for cost-and energy-aware multi-flow mobile data offloading. *IEICE Transactions on Communications*, E101.B(7), 1625–1634.
71. Li, J., Gao, H., Lv, T., & Lu, Y. (2018). Deep reinforcement learning based computation offloading and resource allocation for MEC. In *IEEE Wireless Communications and Networking Conference, WCNC* (pp. 1–6). IEEE.
72. Chen, X., Zhang, H., Wu, C., Mao, S., Ji, Y., & Bennis, M. (2018). Performance optimization in mobile-edge computing via deep reinforcement learning. In *IEEE 88th Vehicular Technology Conference, VTC-Fall* (pp. 1–6). IEEE.
73. Chen, X., et al. (2018). Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning. *IEEE Internet of Things Journal*, 6(3), 4005–4018.
74. Zhu, J., Song, Y., Jiang, D., & Song, H. (2017). A new deep-Q-learning-based transmission scheduling mechanism for the cognitive internet of things. *IEEE Internet of Things Journal*, 5(4), 2375–2385.
75. Gadaleta, M., Chiariotti, F., Rossi, M., & Zanella, A. (2017). D-DASH: A deep Q-learning framework for DASH video streaming. *IEEE Transactions on Cognitive Communications and Networking*, 3(4), 703–718.
76. Naparstek, O., & Cohen, K. (2017). Deep multi-user reinforcement learning for dynamic spectrum access in multichannel wireless networks. In *GLOBECOM 2017–2017 IEEE Global Communications Conference* (pp. 1–7). IEEE.
77. Li, R., Zhao, Z., Chen, X., Palicot, J., & Zhang, H. (2014). TACT: A transfer actor-critic learning framework for energy saving in cellular radio access networks. *IEEE Transactions on Wireless Communications*, 13(4), 2000–2011.
78. Sun, Y., Peng, M., & Mao, S. (2019). Deep reinforcement learning-based mode selection and resource management for green fog radio access networks. *IEEE Internet of Things Journal*, 6(2), 1960–1971.
79. Parera, C., Redondi, A. E. C., Cesana, M., Liao, Q., & Malanchini, I. (2019). Transfer learning for channel quality prediction. In *2019 IEEE international symposium on Measurements & Networking, M & N* (pp. 1–6). IEEE.
80. Zappone, A., Di Renzo, M., Debbah, M., Lam, T. T., & Qian, X. (2019). Model aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks for wireless system optimization. *IEEE Vehicular Technology Magazine*, 14(3), 60–69.
81. Dong, R., She, C., Hardjawana, W., Li, Y., & Vucetic, B. (2020). Deep learning for radio resource allocation with diverse quality-of-service requirements in 5G. *IEEE Transactions on Wireless Communications*, 20(4), 2309–2324.

82. Zhao, Q., Grace, D., Vilhar, A., & Javornik, T. (2015). Using k-means clustering with transfer and Q learning for spectrum, load and energy optimization in opportunistic mobile broadband networks. In *2015 International symposium on wireless communication systems, ISWCS* (pp. 116–120). IEEE.
83. Parera, C., et al. (2020). Transfer learning for multi-step resource utilization prediction. In *IEEE 31st Annual international symposium on personal, indoor and mobile radio communications* (pp. 1–6). IEEE.
84. Zeng, Q., et al. (2020). Traffic prediction of wireless cellular networks based on deep transfer learning and cross-domain data. *IEEE Access*, 8, 172387–172397.
85. Zhang, C., et al. (2019). Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data. *IEEE Journal on Selected Areas in Communications*, 37(6), 1389–1401.
86. Nagaraja, B. B., & Nagananda, K. G. (2015). Caching with unknown popularity profiles in small cell networks. In *IEEE global communications conference, GLOBECOM* (pp. 1–6). IEEE.
87. Hou, T., et al. (2017). Proactive content caching by exploiting transfer learning for mobile edge computing. In *GLOBECOM 2017–2017 IEEE Global Communications Conference* (pp. 1–6). IEEE.
88. Bharath, B. N., Nagananda, K. G., & Poor, H. V. (2016). A learning-based approach to caching in heterogenous small cell networks. *IEEE Transactions on Communications*, 64(4), 1674–1686.
89. Liu, K., et al. (2017). Toward low-overhead fingerprint-based indoor localization via transfer learning: Design, implementation, and evaluation. *IEEE Transactions on Industrial Informatics*, 14(3), 898–908.
90. Zou, H., et al. (2016). A transfer kernel learning based strategy for adaptive localization in dynamic indoor environments: Poster. In *Proceedings of the 22nd Annual international conference on mobile computing and networking* (pp. 462–464).
91. Sun, Z., et al. (2008). Adaptive localization through transfer learning in indoor wi-fi environment. In *Seventh international conference on machine learning and applications* (pp. 331–336). IEEE.
92. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *2017 International Conference on Artificial Intelligence and Statistics, AISTATS* (pp. 1–10). PMLR.
93. Qian, Y., et al. (2019). Privacy-aware service placement for mobile edge computing via federated learning. *Information Sciences*, 505, 562–570.
94. Samarakoon, S., et al. (2018). Federated learning for ultra-reliable low latency V2V communications. In *2018 IEEE global communications conference, GLOBECOM* (pp. 1–7). IEEE.
95. Ye, D., et al. (2020). Federated learning in vehicular edge computing: A selective model aggregation approach. *IEEE Access*, 8, 23920–23935.
96. Dunjko, V., Taylor, J. M., & Briegel, H. J. (2017). Advances in quantum reinforcement learning. In *2017 IEEE International conference on systems, man and, cybernetics, SMC* (pp. 282–287). IEEE.
97. Lu, G., & Zeng, W. H. (2014). Cloud computing survey. *Applied Mechanics and Materials*, 530, 650–661.
98. Chen, J., & Ran, X. (2019). Deep learning with edge computing: A review. *Proceedings of the IEEE*, 107(8), 1655–1674.
99. Kang, Y., et al. (2017). Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. In *2017 International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS* (pp. 615–629). IEEE.
100. Lyu, X., et al. (2021). Distributed online learning of cooperative caching in edge cloud. *IEEE Transactions on Mobile Computing*, 20(8), 2550–2562.
101. Sadeghi, A., Sheikholeslami, F., & Giannakis, G. B. (2018). Optimal and scalable caching for 5G using reinforcement learning of space–time popularities. *IEEE Journal on Selected Topics in Signal Processing*, 12(1), 180–190.

102. Kwak, J., Kim, Y., Lee, J., & Chong, S. (2015). DREAM: Dynamic resource and task allocation for energy minimization in mobile cloud systems. *IEEE Journal on Selected Areas in Communications*, 33(12), 2510–2523.
103. Chen, X. (2015). Decentralized computation offloading game for mobile cloud computing. *IEEE Transactions on Parallel and Distributed Systems*, 26(4), 974–983.
104. Kim, Y., Kwak, J., & Chong, S. (2017). Dual-side optimization for cost-delay tradeoff in mobile edge computing. *IEEE Transactions on Vehicular Technology*, 67(2), 1765–1781.
105. Ding, G., et al. (2018). Spectrum inference in cognitive radio networks: Algorithms and applications. *IEEE Communications Surveys and Tutorials*, 20(1), 150–182.
106. Yun, D. W., & Lee, W. C. (2021). Intelligent dynamic spectrum resource management based on sensing data in space–time and frequency domain. *Sensors*, 21(16), 5261.
107. Shin, D. M., Lim, S. C., & Yang, K. (2012). Mapping selection and code construction for 2m-ary polar-coded modulation. *IEEE Communications Letters*, 16(6), 905–908.
108. Mondelli, M., Hassani, S. H., & Urbanke, R. (2017). Construction of polar codes with sublinear complexity. In *In: 2017 IEEE Int. Symp. Inf* (pp. 1853–1857).
109. Shafin, R., et al. (2020). Artificial intelligence-enabled cellular networks: A critical path to beyond-5G and 6G. *IEEE wireless communications*, 27(2), 212–217.
110. Uwaechia, A. N., & Mahyuddin, N. M. (2019). Spectrum-efficient distributed compressed sensing based channel estimation for OFDM systems over doubly selective channels. *IEEE Access*, 7, 35072–35088.
111. Blue, M. M., Yrjola, S., & Ahokangas, P. (2020). Spectrum management in the 6G era: The role of regulation and spectrum sharing. In *2020 2nd 6G Wirel. Summit. 6G SUMMIT* (pp. 1–5). IEEE.
112. Tariq, F., Khandaker, M. R., Wong, K.-K., et al. (2020). A speculative study on 6G. *IEEE wireless communications*, 27(4), 118–125.

AI Meets SDN: A Survey of Artificial Intelligent Techniques Applied to Software-Defined Networks



Yadunath Pathak, P. V. N. Prashanth, and Ashish Tiwari

1 Introduction

Due to the rising use of smart gadgets such as smartphones, smart watches, smart cars etc., data exchange over the Internet has surged exponentially in recent day [1]. Further, the recent advances in the network technologies such as cloud and edge computing, DevOps, SD-WAN etc. are facilitating such data exchanges by providing increased connectivity between different networking entities. Networks are becoming complex in order to manage a high number of devices and optimize traffic distribution. Typically, a real time network utilizes a variety of hardware devices, runs different protocols to support varied applications. The diverse network architecture presents a variety of difficulties for efficiently planning, controlling, and optimizing networks.

One potential approach to resolving these problems is to add intelligence into networks. An early approach referred to as Knowledge Plane (KP) [2] was proposed to leverage Machine Learning (ML) to automate and add intelligence to the Internet. However, its deployment was not possible due to the distributed nature of traditional network architecture, where a router or switch has control over a small area of a large network. Learning from devices that constitute only a part of the entire network is difficult. The recent networking trend known as Software Defined Networking (SDN) makes the learning task easy from the entire network.

The idea of the SDN is to have a separate control plane and the data plane. The logically centralized control plane (SDN controller) monitors and collects the entire network state to build a global view of the network. Leveraging machine learning algorithms in SDN is beneficial for (a) With the global view of the entire

Y. Pathak (✉) · P. V. N. Prashanth · A. Tiwari

Department of Computer Science & Engineering, Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, India

e-mail: yadunathpathak@cse.vnit.ac.in; prashanthpvn@cse.vnit.ac.in; at@cse.vnit.ac.in

network, the controller can easily collect network data which is the key for applying machine learning algorithms for any network operation, (b) add intelligence to SDN controller using both the past as well as the real time network data, and (c) apply the outcomes of ML algorithms using the programmability of SDN on the network [3, 4].

In this chapter, we review how AI enabled SDN controllers are used for improving network operations and security. First, we provide a background of SDN architecture. Then we review different machine learning (ML) algorithms and analyze different existing works that use these machine learning algorithms when used in conjunction with SDN to achieve optimized solutions in the areas of QoS predictions, optimizing routes, managing resources and enhancing security.. Finally, this chapter discusses the challenges and future directions for applying ML algorithms in SDN.

2 Architecture of SDN

The architecture of SDN comprises three planes- data plane, control plane and the application plane. Figure 1 illustrates a high level overview of the SDN architecture.

1. **Data Plane:** Data plane comprises forwarding devices that include both physical and virtual switches. While physical switches are hardware based, virtual switches are software based (e.g Open vSwitch [7]). Network hardware manufacturers, including HP, Juniper, and Cisco all provide SDN switches. Generally, hardware switches process packets at a faster rate than virtual switches [3]. The switches forward, discard, or alter packets based on the instructions (in the form of flow rules) received from the SDN controller. Southbound interfaces enable communication of flow rules between control and data planes.
2. **Control Plane:** The control plane (CP) has the ability to programme network resources, dynamically change forwarding rules, and provide flexible network administration. The logically centralized SDN controller is the fundamental part of CP. The controller abstracts and provides the network state from data plane to the application plane and delivers customized rules created from the requirements from different network applications to forwarding devices. NOX [8], Floodlight [10], POX [9], Ryu [11], and OpenDayLight [12] are different controller architectures available.

The controller communicates with data plane and application plane using the southbound and northbound interfaces (SBIs and NBIs) respectively. Using SBI, the controller can programmatically control the forwarding devices, receive event notifications and monitoring reports. OpenFlow [6] is the most well-known and widely used open standard SBI today. Others include OVSDB [13], NETCONF [14] etc. Using the abstract network view provided by the controller, applications can change the network behaviors as per their requirements, validate new innovations and efficiently manage networks. The Open Networking Foundation

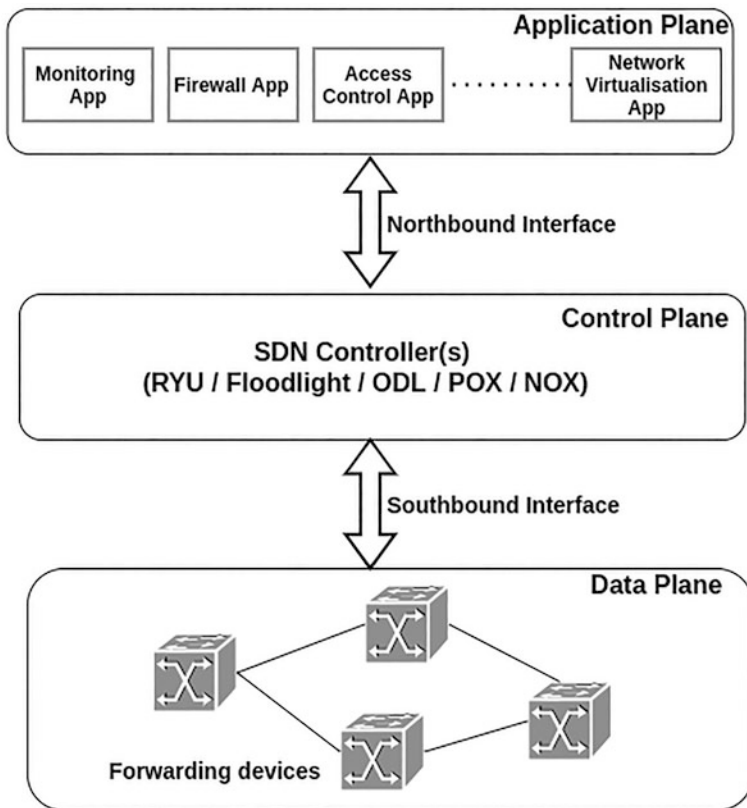


Fig. 1 Overview of SDN architecture

(ONF) is attempting to establish a common information model and the standard NBIs [15].

To efficiently process a large number of flows, large-scale networks are generally divided into smaller network domains. There is a separate controller for each domain. Multiple controllers must communicate via east/west bound interfaces in order to exchange information and give upper-layer apps a comprehensive network perspective. Several distributed control architectures are proposed in the literature [16].

3. **Application Plane:** Application plane is responsible for business management and network optimization. Applications can get network state details from controllers' NBIs. Applications can adjust network behavior based on information and business requirements.

3 Artificial Intelligence in Software-Defined Networking

Broadly speaking, Artificial Intelligence (AI) includes knowledge representation, reasoning, planning, decision making, optimization, machine learning, and meta-heuristic algorithms [5]. The usage of AI has increased across multiple domains in recent days. Several contributions are made by the research community pertaining to the use of machine learning techniques to improve different operational aspects of the SDN paradigm. In the following subsections we provide an overview of different machine learning techniques and then discuss how these techniques are applied to achieve different goals in the context of SDN.

3.1 *Supervised Learning Techniques*

In the supervised learning approaches, prior knowledge is offered to the system. In general, a training dataset consisting of input-output pairs is provided, from which the system learns a mapping function. The trained model thus obtained is used to predict the output when a fresh input is introduced into the system after training [17]. The most popular supervised learning methods used in SDN are briefly explained in the following subsections.

3.1.1 **Support Vector Machines (SVM)**

The SVM technique creates the best decision boundary (called a hyperplane) to separate high dimension space into different classes in order to easily place new data points in the right category [18]. SVM chooses the support vectors that help to create the hyperplane. While linear SVMs classify linearly separable data into two classes using a straight line, non-linear SVMs are used for datasets that cannot be classified using a straight line. SVMs can be used to represent complicated functions and are resistant to overfitting [18].

3.1.2 **K-Nearest Neighbor (K-NN)**

K-NN classifies the sample data using the k-nearest neighbors of the unclassified sample. An unclassified sample will be placed in the class to which the majority of its k-nearest neighbors belong to [19]. Because the primary metric used by the k-NN algorithm is the distance, a number of functions, including Chebyshev, City-block, Euclidean, and Euclidean squared are used to calculate the distance between an unlabeled data and their neighbors.

3.1.3 Decision Trees

Classification problems respond well to decision trees (DT). A data set can be described by a tree-like structure using DT [23]. Discrete or continuous data can be used as input and output. All Boolean functions can be represented using decision trees. Decision trees run a series of tests where the nodes of the decision tree correspond to the tests of a different property from the input [18]. One of the key benefits of DT is its interpretability, or the capacity to comprehend the rationale behind a learning algorithm's output. However, DT experiences over-fitting, which can result in a massive tree when there is no discernible structure in the input data [18].

3.1.4 Artificial Neural Network (ANN)

The key idea behind neural networks comes from the way the human brain works. The brain uses simple parts called neurons to do complex, nonlinear, and parallel calculations. A neural network is a system that is made up of a lot of simple processing units that work together to learn from past experiences [20]. Nodes of an ANN are like the neurons of the human brain. Activation functions are used by these nodes to do nonlinear computations. The sigmoid and hyperbolic tangent functions are the ones that are used most often as activation functions [21]. In an ANN, the nodes are connected to each other by links with different weights similar to how the neurons in the human brain are connected.

An ANN has many layers. The first layer is where information comes in, and the last layer is where information goes out. There are hidden layers also between the first layer and the last layer. Each layer takes input from the previous layer and the overall output is the output of the last layer. Complex models can be trained to make ANNs work better by varying the hidden layers and the nodes in each of these layers.

ANN offers numerous benefits. (1) They easily adapt to the data without explicitly establishing a representational function or distribution in the underlying model [22]. (2) ANNs constitute a universal function approximator [22] capable of approximating any function, and (3) ANNs are nonlinear models, which gives them the ability to express and model complicated interactions. The majority of MLPs are trained via supervised training algorithms. When using too many parameters in a model, neural networks are susceptible to overfitting [18].

3.1.5 Ensemble Methods

Ensemble methods integrate the predictions of various other methods and are mostly employed to enhance the learning algorithms performance [24]. Bagging is a technique for improving classification accuracy by combining different outputs of learned classifiers into one output. The learned classifiers are trained using instances

produced from the original data set by random sampling with replacement [24]. In boosting as opposed to bagging, each of these classifiers depend on the prior classifier's performance and focus on the errors caused by them.

3.2 *Unsupervised Learning*

Unsupervised learning methods are provided with unlabelled data and do not have any prior knowledge [18]. Hence, these approaches aim to find patterns in the given data. The following subsections provide the popular unsupervised learning techniques in use.

3.2.1 **K-Means Clustering**

K-means is a popular and widely used clustering technique. The value of parameter k that represents the count of clusters to be produced is specified to the algorithm. Every data point is assigned to the cluster's nearest centroid. K-means minimizes the distance function between data points and their respective centroids [25]. The procedure for revising the centroids depending on the respective data points is repeated until there is no change in the centroid and the points. The K-means algorithm relies on the initial cluster set. Consequently, an improper choice of k results in poor outcomes [26].

3.2.2 **Self-Organizing Feature Maps (SOFM)**

SOFM [27] maps high-dimensional distributions to low-dimensional representations termed SOFM maps [28]. SOFMs help in recognizing patterns, even the noisy ones. SOFMs are built, reorganized during the training process based on the input data set and then classifies a given (new) input vector by determining the winning neuron or the node [28].

3.2.3 **Hidden Markov Model (HMM)**

Hidden Markov Model (HMM) [29] assumes the system under study is a Markov process with hidden (un-observed) states. The Markov process states that the probability of one state depends only on the probability of the preceding state [30]. Five different entities are specified in the model by the HMM: (a) state set, (b) output alphabet, (c) initial probability, (d) transition probabilities, and (e) observation probability [30]. HMM parameters can be trained using supervised or supervised methods. Baum-Welch method [30] is regarded as the most commonly used unsupervised algorithm.

3.2.4 Semi-Supervised Learning

Semi-supervised learning makes use of both labeled and unlabeled data. Acquiring labeled data is generally costly and also difficult in real world applications while unlabeled data is cheap and easy to acquire. Further, using unlabeled data during training improves model performance. Pseudo labeling [31, 32] is one of semi supervised learning techniques. In pseudo labeling, the model is trained with labeled data initially. Then, the prediction of pseudo labels of unlabeled data is done using the learned model. Finally, retrain the model using labeled and pseudo-labeled data. Expectation Maximization, transductive SVM, and co-training are some of the semi-supervised learning methods.

3.2.5 Restricted Boltzmann Machine (RBM) and Deep Belief Networks (DBN)

RBMs are stochastic ANNs with two layers - input and hidden [33]. In RBMs, unlike simple Boltzmann machines, there is a connection between every neuron of the input layer to all the hidden neurons and vice versa. However, there is no connection between any two neurons of the same layer. The bias unit connects all visible and hidden neurons. Deep belief networks (DBNs) need RBMs for feature extraction [33].

DBN is a kind of generative ANNs where multiple RBMs may be layered to create a deep learning model [34]. DBNs obtain hierarchical representationSDNs of training data and rebuild input data. DBN works effectively because of the training that happens layer on layer. Every layer is viewed as an RBM that is trained on top of its preceding layer. DBNs can be used to find hierarchical features [34].

3.3 Reinforcement Learning (RL) Techniques

The RL system acquires knowledge using a set of environmental reinforcements. An RL system has an agent, a set of states and a set of actions. The agent performs actions to move from one state to another. For every action taken, the agent gets an intermediate reward that represents how good the action is. The agent's ultimate goal is to learn the optimized behavioral policy, which is nothing but a map from state space to action space, in order to achieve higher long-term reward. The agent can then choose the best action based on the policy. The value function of RL is used to determine the long term reward of an action for a corresponding state. When RL is applied to SDN, the controller takes the role of an agent that performs action on the network considering it as the environment.

3.3.1 Q-Learning

Q-learning is a type of reinforcement learning that enables agents to operate optimally in a controlled set of markovian domains without needing to construct maps for the domains [35]. The objective of the agent is to determine optimal policy that aims to maximize the total dis-counted reward, commonly known as Q, for executing a specific action in a specific state. Further, Q-learning is categorized as a type of dynamic programming technique that is incremental in nature since it determines the optimized policy in stages [35].

3.3.2 Deep Reinforcement Learning

Deep reinforcement learning (DRL) is a solution to the problems of RL that include lower rate of convergence to optimized policy, inability to scale up with high dimensional state space and action space [36]. DRL relies mostly on deep NNs to approximate the optimized policy.

DRL leverages the deep NN approximation function to approximate the value function. DRL estimates long term reward from a given state action pairs after training deep NNs. The result of the estimation can help the agent decide the best action.

4 AI Enabled SDN Controllers

Machine learning approaches add intelligence to the SDN controllers thereby providing it the ability to analyze network information, optimize network resource utilization, and automate network service provisioning. This section reviews different machine learning techniques proposed to handle the issues in SDN corresponding to classification of different types of traffic, optimized routing decisions, predicting QoS guarantees, management of network resources as well as security.

4.1 AI-Enabled Traffic Classification in SDN

An SDN controller obtains the global view of the network and collects flow level statistics of the network. Using these flow statistics an SDN controller can efficiently classify the network traffic into different traffic flow categories. Using this information in conjunction with ML approaches, SDN controllers are able to achieve network management at flow level thereby enabling the network operators to handle diverse applications and distribute resources effectively.

In general, traffic is classified using (a) port numbers, (b) deep packet Inspection (DPI), and (c) machine learning approaches [3]. Mostly the applications are

determined by UDP and TCP port numbers. Many applications used TCP port 80 for HTTP in the past. However, the port-based traffic classification method is no longer viable because most applications now use dynamic ports. DPI matches traffic flow payloads to predetermined patterns to identify applications. Regular expressions are used to define patterns. DPI-based classification is accurate. However, it has some flaws (a) DPI only recognises available patterns. Updating patterns becomes difficult and impractical due to a diverse and large number of applications. (b) DPI requires checking all traffic flows, which is computationally expensive, and (c) DPI can't classify encrypted traffic. ML-based techniques have the ability to recognise encrypted communication and have less computing costs. ML-based techniques have been investigated extensively. The application of ML and SDN for elephant flow-aware, application-aware, and QoS-aware traffic classification are reviewed below.

Typically, 80% of data center traffic corresponds to mice flows. Elephant flows carry most bytes [40]. Therefore, identifying elephant flows helps control data center traffic. Glick and Rastegarfar [41] analyzes data center traffic flow scheduling. The authors use machine learning techniques for classifying elephant flows and mice flows at the network edge first and then use the result of the classification in a centralized SDN controller to optimize traffic flows. In [42], a two step learning approach for detecting elephant flows in SDN is proposed. In step 1, measurement of head packet is used to identify elephant flows from mice flows. In step 2, a decision tree is utilized to determine if suspected elephant flows are elephant flows.

Amaral et al. [37] examined application aware network traffic classification. The authors collect traffic data using a simple OpenFlow-based SDN solution. Then, classifier algorithms are used to classify the identified flows into different applications. MultiClassifier combines ML-based and DPI-based classifiers to detect applications [43]. ML-based classifiers are used to classify fresh flows. If the ML-classifier's result is reliable enough, it is used. Otherwise, the approach falls back to classification using DPI. Rossi and Valenti [44] classifies UDP applications. SVM algorithm is used to classify UDP traffic based on Netflow data. Simulations showed that the classification engine based on SVM achieved accuracy of over 90%. Qazi et al. [38] classifies mobile application traffic. To identify mobile apps, Atlas is proposed. Decision tree is used in the approach and is trained using the data acquired using the crowdsourcing approach. Simulation results show that Atlas's categorization accuracy for 40 top Google Play apps is over 94%. Deep NN are utilized to identify the mobile applications in [45]. Experimental traffic is collected and features including (a) destination address, (b) destination port, (c) protocol type, (d) TTL, and (e) packet size are used for training an eight layer deep neural network model. Simulations show the trained model can identify 200 mobile apps with 93.5% accuracy.

QoS aware traffic classification identifies traffic flows depending upon the QoS class of the traffic. With the proliferation of Internet applications, it's hard to recognise them all. Applications can be categorized by their QoS needs (delay, jitter, loss rate). QoS classes include several uses. It's a better technique to classify traffic flows by QoS. Using semi-supervised learning and DPI, [39] proposes a QoS-aware

traffic classification system. DPI labels application traffic flows. Semi-supervised learning algorithms use the labeled training dataset to classify unknown application traffic flows. Known and unknown application traffic flows are divided into QoS classes. Simulations show the system's categorization accuracy is over 90%.

4.2 AI-Enabled Routing in SDN

In the SDN paradigm, the SDN controller can instruct the switches (in the form of flow rules) to route packets following a specific path. The inefficient routing decisions made by the SDN controller can overburden the links and increase E2E transmission delay, affecting SDN performance. Optimizing traffic flow routing is a major research issue. Shortest path forwarding is the widely used routing approach. However, the algorithm does not guarantee the best utilization of network resources. Heuristic approaches such as genetic algorithm based routing and others are used for route optimization but the cost of computing paths for a flow is high [46, 47]. ML algorithms when applied for making routing decisions can provide a near optimal solution. Since a mathematical model is not required, the route optimization problem boils down to decision making tasks, thereby, making the RL an efficient approach for route optimizations.

Sendra et al. [48] offers RL-based distributed intelligent routing in SDN. The suggested routing system selects appropriate data transmission paths according to network status. Francois et al. [49, 50] examine inter-data center SDN routing optimization. Using random NN and RL, a centralized Cognitive Routing Engine (CRE) finds optimal overlay pathways between geographically-dispersed data centers. The proposed CRE works effectively in chaotic conditions based on random NN and RL. Lin et al. [51] discusses multi-layer hierarchical SDN routing optimization. Using the RL algorithm, QoS-aware Adaptive Routing (QAR) enables time-efficient adaptive packet forwarding. Traffic categories and user applications determine the QoS-aware routing path. López-Raventós et al. [52] optimizes routing with DRL. The DRL model selects the appropriate source-destination routing paths given the traffic matrix to reduce network delay.

4.3 AI-Enabled Traffic Predictions in SDN

Predicting network traffic is an essential routing optimization research topic. Traffic prediction analyzes previous traffic data to predict traffic volume [53]. The SDN controller makes traffic routing decisions based on traffic prediction results and delivers proactive routing rules to data plane forwarding devices to guide future traffic flow routing. So, the SDN controller can prevent traffic congestion. Traffic prediction can also help provide QoS-improving network resources.

Alvizu et al. [54] proposes a load-balancing approach to optimize load on the path. SDN controller selects 4 traffic flow features – packet loss rate, packet loss rate, bandwidth usage ratio and latency to predict the path load using a neural network model. New traffic flows will use the least-loaded path. NeuTM, an Long Short-Term Memory (LSTM) based framework, predicts network traffic matrix [55]. LSTM is trained using GEANT backbone traffic [56]. The LSTM model converges quickly and performs well in simulations.

4.4 AI-Enabled Prediction of Quality of Service/Experience

Network operators employ QoS factors including rate of packet loss, latency, throughput and jitter to assess performance of the network. User perception and satisfaction are also crucial for service providers as multimedia technologies become more mainstream in recent days. QoE measures user satisfaction with a service. Service providers can deliver quality services based on QoS/QoE predictions to boost customer satisfaction and stop customer attrition. SDN controllers have the ability to obtain per-port and per-flow switch statistics for ML QoS/QoE prediction.

Carner et al. [57] seeks to automatically predict delays in the network given the traffic load and overlay routing scheme. Two models (M/M/1 and neural) are presented to estimate delay. The NN-based estimator outperforms the M/M/1 model in delay estimation accuracy. Jain et al. [58] proposes a two-phase SDN study to improve QoS prediction. First, decision trees find correlations in between QoS parameters and KPIs. Thereafter, a linear regression ML technique is used to determine each KPI's quantitative influence. The suggested technique predicts traffic congestion and improves QoS.

Quality of Experience measures user satisfaction subjectively. Mean Opinion Score (MOS) is a popular QoE metric [59, 60] that has five QoE levels including bad, poor, fair, good and excellent. The values of QoE are commonly obtained using subjective methods, where users rate a service's quality. Subjective methods used for QoE evaluation are time-consuming. Hence, to acquire QoE numbers in real time, it's vital to understand the effect of QoS parameters on QoE values. Machine learning is a good way to learn the QoS-QoE relationship. As QoE values are discrete, predicting them is a classification problem. Therefore, supervised learning is optimal for QoE prediction. Letaifa et al. [59] predicts QoE for video streaming in SDN. MOS is estimated using network metrics (RTT, jitter, bandwidth, delay). The SDN controller alters the video characteristics (e.g., bitrate, resolution, frame rate, and resolution) to optimize QoE. Abar et al. [60] proposed to use four ML methods to predict QoE based on the characteristics of video quality. These algorithms are evaluated using Pearson correlation coefficient and Root-Mean-Square-Error.

4.5 *AI-Enabled Resource Management in SDN*

Improving network performance requires efficient network resource management. SDN separates the control and data planes, allowing the controller with a global view of the network to programme the entire network. SDN enables resource management by maximizing the network resource utilization. The major resources in the data plane of a network are the (a) network resources including bandwidth, power consumption of the devices, and spectrum and (b) the computing resources that are required for computation intensive applications such as face recognition and augmented reality [3, 5].

In multi-tenant SDN networks where resources are shared among multiple tenants, the allocation of data plane resources by maximizing their utility is a challenging issue. Network hypervisors introduced in multi-tenant SDN networks enable tenants to use their own controllers to control data-plane resources. Network hypervisors (e.g flowvisor, openvrtex etc) have the task of processing control traffic of the traffic between the data plane and control plane, yet have limited processing resources. The question of managing network hypervisors' (limited) resources across several tenants to ensure data and control plane connection is a significant research subject [3]. ML methods optimize resource allocation on network hypervisors in this case.

In [61], a resource monitoring tool was proposed to monitor network hypervisor CPU utilization and hvbench, a benchmarking tool measures control message rate. The obtained data trains three regression learning models to map CPU usage to control message rate. Using the measured rate of control messages, trained models can predict if hypervisors are burdened in real time. Overloaded network hypervisors affect tenant control message processing.

Controller placement is another challenging issue that has to be addressed as the distance between the controller and the forwarding devices affect the network performance. In [62, 63], neural network, decision tree, and logistic regression are used to find the ideal controller placement positions. Traffic distribution is input to supervise learning algorithms, while heuristic methods produce controller placement options. After the training process, the model is able to predict real-time controller placement. Bendriss et al. [64] examines Service Level Agreement (SLA) management in SDN. First, LSTM [65] a commonly used recurrent NNs calculates the system features at time $t + 1$ based on their values at time t . Decision tree algorithm uses forecasted feature values to estimate the most likely SLA Violation. Then, proactive management prevents SLA violation. Saurav et al. [72] suggested employing machine learning algorithms based on past network attack data to predict malicious connections and attack targets. Based on historical data, the authors applied four machine learning algorithms-C4.5, Bayesian Network, Decision Table, and Naive-Bayes.

4.6 *Meta-heuristic Algorithms in SDN*

Yu et al. [74] proposed GA-SDN, a genetic algorithm-based routing system for improving video delivery over SDNs. Bouet et al. [75] provided a technique for optimal deep packet inspection deployment in NFV-SDNs. Hu et al. [76] presented a strategy for maximizing network reliability by addressing controller placement problem in SDNs. Sathya et al. [73] presented a binary bat algorithm in an SDN-based intrusion detection system. Parsaei et al. [77] studied QoS for telesurgery. The suggested approach periodically collects network status facts and uses Ant Colony Optimization to calculate the optimum surgeon-to-patient path. The authors are suggested to go through [5] for a more comprehensive review of different meta-heuristic algorithms used in SDN for addressing various challenges.

5 Challenges of Applying AI Techniques in SDN

Despite current progress in SDN numerous research difficulties remain until a fully intelligent SDN is widely implemented. The following subsections discuss these problems and possible future research directions:

5.1 *Lack of Training Datasets*

Sufficient training datasets are needed to increase machine learning models' estimation or classification accuracy [2, 3, 5]. The relationship between size of the training dataset, network properties, and machine learning model performance should be studied. Machine learning algorithms rely on high quality training datasets. It's hard to find high-quality annotated network flow samples across applications [37]. Public datasets are a frequent solution in machine learning applications. Similarly, AI networking datasets must be published.

5.2 *Handling Large Networks with Multi-controllers*

With a single SDN controller controlling the entire control plane, scaling issues arise due to the controller's computation limitation [66]. In a hierarchical, distributed multi-controller platform [51] there is a root controller at the top of the hierarchy and local controllers that connect to the root controller. The root controller can access all switches and sees the entire network. Each local controller can only control part of a domain's switches and has network status information. Optimal inter-domain and intra-domain traffic routing is an issue that needs to be addressed.

SDN reliability is another challenge due to the controller's single point of failure. A set of SDN controllers can be used to overcome the dependability issue. A cluster of controllers processes flow table update requests. The question of choosing an optimal controller for updating flow tables is an interesting area of research. ML algorithms are a possible solution for handling these situations to optimize the flow traffic and minimize resource utilizations.

5.3 Addressing Security Aspects of SDN

Data and control planes separation minimizes network device complexity and allows flexible network administration. Since data plane switches are dumb, they provide the controller raw data packets. This practice provides a major vulnerability that attackers can use to overload the controller with flow requests. SDN controllers utilize ML-based anomaly detection to detect network assaults. Malicious attackers constantly build new attacks to circumvent anomaly detection. Due to new assaults, utilizing past data for training ML models may not always be successful. Efficient ML algorithms need to be applied to stop these attacks.

5.4 Partial SDN Deployment

SDN's deployment requires complete SDN-aware network infrastructure. Incremental deployment is a solution. SDN controllers and SDN enabled switches are deployed incrementally in the traditional network, and only a part of network traffic is regulated. The SDN controller may exchange bandwidth, link weights, and topological information with traditional network nodes also. So, the SDN controller can obtain network data. Using these switch flow statistics data, machine learning algorithms may construct models for resource allocation optimization and traffic engineering.

6 Knowledge-Defined Networking

In this section, we review recent works related to KDN, a concept that is designed to use both ML and SDN together. Mestres et al. [67] developed ML methods in KDN for controlling and operating networks. Pham et al. [68], used deep RL with CNN to determine QoS-based routing decisions in complicated networks. Lu et al. [69], used deep learning techniques to generate AI modules to regulate energy consumption and scalability in data center networks. These modules are used to accurately predict traffic volume, latency, and hardware demands. In [70], the authors introduced a new 5G network architecture and solution using SDN, KDN, and NFV, using ML

in KDN. Due to massive device connectivity, self-driving networks using KDN and network telemetry were proposed in [71]. However, there are also some problems associated with these works as to how KDN is used in general for KDN to be completely operational.

7 Conclusion

This chapter provided an overview of SDN architecture and the machine learning techniques in use today. The chapter reviewed different machine learning techniques applied to the context of SDN for enhancing the network management. Specifically, this chapter reviewed different machine learning techniques applied for traffic classification, predicting traffic, predicting quality of service/experience, resource management and routing. Further, this chapter highlighted different challenges and possible future directions for applying machine learning techniques to SDN.

References

1. Gartner. (2017, January). *Forecast: Internet of things – Endpoints and associated services, worldwide, 2016* (Technical report). Gartner.
2. Clark, D. D., Partridge, C., Ramming, J. C., & Wroclawski, J. T. (2003). A knowledge plane for the Internet. In *Proceedings of ACM SIGCOMM* (pp. 3–10).
3. Xie, J., et al. (2019). A survey of machine learning techniques applied to Software Defined Networking (SDN): Research issues and challenges. *IEEE Communications Surveys & Tutorials*, 21(1), 393–430. <https://doi.org/10.1109/COMST.2018.2866942>
4. Xu, G., Mu, Y., & Liu, J. (2017, Oct). Inclusion of artificial intelligence in communication networks and services. *ITU Journal: ICT Discoveries*, (1), 1–6.
5. Latah, M., & Toker, L. (2019). Artificial intelligence enabled software-defined networking: A comprehensive overview. *IET Networks*, 8(2), 79–99.
6. McKeown, N., Anderson, T., Balakrishnan, H., et al. (2008). OpenFlow: Enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2), 69–74.
7. Open vSwitch. Available: <https://www.openvswitch.org/>
8. Gude, N., et al. (Jul. 2008). NOX: Towards an operating system for networks. *SIGCOMM Computer Communication Review*, 38(3), 105–110.
9. McCauley, M. (2013). *About Pox*. [Online]. Available: <http://www.noxrepo.org/pox/about-pox/>
10. Floodlight. (2012). *Project floodlight open source software for building software defined networks*. [Online]. Available: <http://www.projectfloodlight.org/>
11. Ryu. (2013). *Ryu SDN framework*. [Online]. Available: <http://osrg.github.io/ryu/>
12. Medved, J., Varga, R., Tkacik, A., & Gray, K. (2014, June). OpenDaylight: Towards a model-driven SDN controller architecture. In *Proceedings of IEEE WoWMoM* (pp. 1–6).
13. Pfaff, B., & Davie, B.. (2013, Dec). *The open vSwitch database management protocol*. [Online]. Available: <https://rfc-editor.org/rfc/rfc7047.txt>
14. Enns, R., Bjorklund, M., & Schoenwaelder, J.. (2011, June). *Network Configuration protocol (NETCONF)*. [Online]. Available: <http://www.rfc-editor.org/rfc/rfc6241.txt>
15. Open Networking Foundation. (2016). *Common information model overview. V1.2*. [Online]. Available: <https://www.opennetworking.org/>

16. Koponen, T. et al. (2010). Onix: A distributed control platform for large-scale production networks. In *Proceedings of OSDI* (vol. 10, pp. 1–6).
17. Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. In *Proceedings of Emerging Artificial Intelligence Applications in Computer Engineering* (Vol. 160, pp. 3–24).
18. Russell, S., & Norvig, P. (1995). *Artificial intelligence (A modern approach)* (3rd ed., 1152 p). Prentice Hall.
19. Cover, T., & Hart, P. (1967, Jan). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, IT-13*(1), 21–27.
20. Haykin, S. (1994). *Neural networks: A comprehensive foundation*. Prentice-Hall.
21. Haykin, S. (2004). A comprehensive foundation. *Neural Networks*, 2, 41.
22. Zhang, G. P. (2000). Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4), 451–462.
23. Negnevitsky, M. (2005). *Artificial intelligence – A guide to intelligent systems* (2nd ed., 415 p). Addison-Wesley.
24. Rokach, L. (2008). Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Computational Statistics & Data Analysis*, 53(12), 4046–4072.
25. Khan, S. S., & Ahmad, A. (2004). Cluster center initialization algorithm for K-means clustering. *Pattern Recognition Letters*, 25(11), 1293–1302.
26. Zhang, C., Xia, S. (2009, Jan). K-means clustering algorithm with improved initial center. In *IEEE Second International Workshop on Knowledge Discovery and Data Mining* (pp. 790–792).
27. Kohonen, T. (1988). The self-organizing map. *Neurocomputing*, 21(1–3), 1–6.
28. Phan, T. V., Bao, N. K., & Park, M. (2017). Distributed-SOM: A novel performance bottleneck handler for large-sized software-defined networks under flooding attacks. *Journal of Network and Computer Applications*, 91, 14–25.
29. MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley symposium on mathematical statistics and probability* (pp. 281–297).
30. Fan, Z., Xiao, Y., Nayak, A., & Tan, C. (2017). An improved network security situation assessment approach in software defined networks. *Peer-to-Peer Networking and Applications*.
31. Lee, D.-H. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Proceedings of Workshop Challenges in Representation Learning (ICML)* (Vol. 3, p. 2).
32. Wu, H., & Prasad, S. (Mar. 2018). Semi-supervised deep learning using pseudo labels for hyperspectral image classification. *IEEE Transactions on Image Processing*, 27(3), 1259–1270.
33. Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials*.
34. Zhang, Q., Yang, L. T., Chen, Z., & Li, P. (2018). A survey on deep learning for big data. *Information Fusion*, 42, 146–157.
35. Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3–4), 279–292.
36. Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6), 26–38.
37. Amaral, P. et al. (2016, Nov). Machine learning in software defined networks: Data collection and traffic classification. In *Proceedings of IEEE ICNP* (pp. 1–5).
38. Qazi, Z. A. et al. (2013). Application-awareness in SDN. In *Proceedings of ACM SIGCOMM* (pp. 487–488).
39. Wang, P., Lin, S.-C., & Luo, M. (2016, June/July). A framework for QoS-aware traffic classification using semi-supervised machine learning in SDNs. In *Proceedings of IEEE SCC* (pp. 760–765).
40. Benson, T., Akella, A., & Maltz, D. A. (2010). Network traffic characteristics of data centers in the wild. In *Proceedings of ACM IMC* (pp. 267–280).

41. Glick, M., & Rastegarfar, H. (2017, July). Scheduling and control in hybrid data centers. In *Proceedings of IEEE PHOSSST* (pp. 115–116).
42. Xiao, P., Qu, W., Qi, H., Xu, Y., & Li, Z. (2015, Mar). An efficient elephant flow detection with cost-sensitive SDN. In *Proceedings of IEEE INISCom* (pp. 24–28).
43. Li, Y., & Li, J. (2014, Nov) MultiClassifier: A combination of DPI and ML for application-layer classification in SDN. In *Proceedings of IEEE ICSAI* (pp. 682–686).
44. Rossi, D., & Valenti, S. (2010). Fine-grained traffic classification with Netflow data. In *Proceedings of ACM IWCMC* (pp. 479–483).
45. Nakao, A., & Du, P. (2018). Toward in-network deep machine learning for identifying mobile applications and enabling application specific network slicing. *IEICE Transactions on Communications, E101.B(7)*, 1536–1543.
46. Yanjun, L., Xiaobo, L., & Osamu, Y. (2014, Sept). Traffic engineering framework with machine learning based meta-layer in software-defined networks. In *Proceedings of IEEE ICNDC* (pp. 121–125).
47. Azzouni, A., Boutaba, R., & Pujolle, G. (2017). NeuRoute: Predictive dynamic routing for software-defined networks. *arXiv preprint arXiv:1709.06002*.
48. Sendra, S., Rego, A., Lloret, J., Jimenez, J. M., & Romero, O. (2017, May). Including artificial intelligence in a routing protocol using software defined networks. In *Proceedings of IEEE ICC workshops* (pp. 670–674).
49. Francois, F., & Gelenbe, E. (2016, Sep). Optimizing secure SDN-enabled inter-data center overlay networks through cognitive routing. In *Proceedings of IEEE MASCOTS* (pp. 283–288).
50. Francois, F., & Gelenbe, E. (2016, May). Towards a cognitive routing engine for software defined networks. In *Proceedings of IEEE ICC* (pp. 1–6).
51. Lin, S. C., Akyildiz, I. F., Wang, P., & Luo, M. (2016, June/July). QoS-aware adaptive routing in multi-layer hierarchical software defined networks: A reinforcement learning approach. In *Proceedings of IEEE SCC* (pp. 25–33).
52. López-Raventós, Á., Wilhelmi, F., Barrachina-Muñoz, S., & Bellalta, B. (2018). Machine learning and software defined networks for high-density WLANs. *arXiv preprint arXiv:1804.05534*.
53. Alvizu, R., Troia, S., Maier, G., & Pattavina, A. (2017, Sep). Matheuristic with machine-learning-based prediction for software-defined mobile metrocore networks. *IEEE/OSA Journal of Optical Communications and Networking*, 9(9), D19–D30.
54. Chen-Xiao, C., & Ya-Bin, X. (2016). Research on load balance method in SDN. *International Journal of Grid and Distributed Computing*, 9(1), 25–36.
55. Azzouni, A., & Pujolle, G. (2017). NeuTM: A neural network-based framework for traffic matrix prediction in SDN. *arXiv preprint arXiv:1710.06799*.
56. GEANT. Available: https://www.geant.org/Projects/GEANT_Project_GN4
57. Carner, J., Mestres, A., Alarcón, E., & Cabellos, A. (2017). Machine learning- based network modeling: An artificial neural network model vs a theoretical inspired model. In *Proceedings of IEEE ICUFN* (pp. 522–524).
58. Jain, S., Khandelwal, M., Katkar, A., & Nygate, J. (2016, Oct/Nov). Applying big data technologies to manage QoS in an SDN. In *Proceedings of IEEE CNSM* (pp. 302–306).
59. Letaifa, A. B. (2017). Adaptive QoE monitoring architecture in SDN networks: Video streaming services case. In *Proceedings of IEEE IWCMC* (pp. 1383–1388).
60. Abar, T., Letaifa, A. B., & Asmi, S. E. (2017). Machine learning based QoE prediction in SDN networks. In *Proceedings of IEEE IWCMC* (pp. 1395–1400).
61. Sieber, C., Basta, A., Blenk, A., & Kellerer, W. (2016, June). Online resource mapping for SDN network hypervisors using machine learning. In *Proceedings of IEEE NETSOFT* (pp. 78–82).
62. He, M., Kalmbach, P., Blenk, A., Kellerer, W., & Schmid, S. (2017, Oct). Algorithm-data driven optimization of adaptive communication networks. In *Proceedings of IEEE ICNP* (pp. 1–6).
63. Blenk, A., Kalmbach, P., Kellerer, W., & Schmid, S. (2017). O’Zapft is: Tap your network algorithm’s big data!. In *Proceedings of ACM Big-DAMA* (pp. 19–24).

64. Bendriss, J., Yahia, I. G. B., & Zeghlache, D. (2017, Mar). Forecasting and anticipating SLO breaches in programmable networks. In *Proceedings of IEEE ICIN* (pp. 127–134).
65. Li, X., & Wu, X. (2015, Apr). Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. In *Proceedings of IEEE ICASSP* (pp. 4520–4524).
66. Kreutz, D., et al. (2015, Jan). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1), 14–76.
67. Mestres, A., Rodriguez-Natal, A., Carner, J., Barlet-Ros, P., Alarcón, E., Solé, M., et al. (2017). Knowledge-defined networking. *ACM SIGCOMM Computer Communication Review*, 47(3), 2–10.
68. Pham, T. A. Q., Hadjadj-Aoul, Y., & Outtagarts, A. (2018). Deep reinforcement learning based QoS-aware routing in knowledge-defined networking. In: *International conference on heterogeneous networking for quality, reliability, security and robustness* (pp. 14–26). Springer.
69. Lu, W., Liang, L., Kong, B., Li, B., & Zhu, Z. (2020). AI-assisted knowledge-defined network orchestration for energy-efficient data center networks. *IEEE Communications Magazine*, 58(1), 86–92.
70. Careglio, D., Spadaro, S., Cabellos, A., Lazaro, J., Perelló, J., Barlet, P., et al. (2018). ALLIANCE project: Architecting a knowledge-defined 5G-enabled network infrastructure. In: *2018 20th international conference on transparent optical networks* (pp. 1–6). IEEE.
71. Hyun, J., Van Tu, N., & Hong, J. W.-K. (2018). Towards knowledge-defined networking using in-band network telemetry. In: *NOMS 2018–2018 IEEE/IFIP network operations and management symposium* (pp. 1–7). IEEE.
72. Nanda, S., et al. (2016) Predicting network attack patterns in SDN using machine learning approach. In *2016 IEEE conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. IEEE.
73. Sathya, R., & Thangarajan, R. (2015, Feb). Efficient anomaly detection and mitigation in software defined networking environments. In *Proceedings of 2nd International Conference on Electronics and Communication Systems (ICECS)* (pp. 479–484).
74. Yu, Y. S., & Ke, C. H. (2017). Genetic algorithm-based routing method for enhanced video delivery over software defined networks. *International Journal of Communication Systems*, 31(1), e3391.
75. Bouet, M., Leguay, J., & Conan, V. (2013, Nov). Cost-based placement of virtualized deep packet inspection functions in SDN. In *Proceedings of IEEE military communications conference* (pp. 992–997).
76. Hu, Y., Wendong, W., Gong, X., Que, X., & Shiduan, C. (2013, May). Reliability-aware controller placement for software-defined networks. In *IFIP/IEEE international symposium on integrated network management (IM 2013)* (pp. 672–675).
77. Parsaei, M. R., Mohammadi, R., & Javidan, R. (2017). A new adaptive traffic engineering method for telesurgery using ACO algorithm over software defined networks. *European Research in Telemedicine/La Recherche Européenne en Telemedecine*, 6(3–4), 173–180.