SMART FRAMEWORK FOR ENERGY MANAGEMENT SYSTEM IN EDGE-OF-THINGS

Thesis Submitted for the Award of the Degree of

DOCTOR OF PHILOSOPHY

in

Computer Applications

By

Rajeev Kanday

Registration Number : 41800129

Supervised By

Dr. Ramandeep Singh (14105)

School of Computer Science and Engineering (Professor)

Lovely Professional University

LOVELY PROFESSIONAL UNIVERSITY, PUNJAB

2024

DECLARATION

I, hereby declared that the presented work in the thesis entitled "Smart Framework for Energy Management System in Edge-of-Things" in fulfilment of degree of **Doctor of Philosophy (Ph. D.)** is outcome of research work carried out by me under the supervision of Dr. Ramandeep Singh, working as Professor, in the School of Computer Science and Engineering of Lovely Professional University, Punjab, India. In keeping with general practice of reporting scientific observations, due acknowledgements have been made whenever work described here has been based on findings of other investigator. This work has not been submitted in part or full to any other University or Institute for the award of any degree.

(Signature of Scholar)

Name of the scholar: Mr. Rajeev Kanday

Registration No.:41800129

Department/school: School of Computer Applications

Lovely Professional University,

Punjab, India

CERTIFICATE

This is to certify that the work reported in the Ph. D. thesis entitled "Smart Framework for Energy Management System in Edge-of-Things" submitted in fulfillment of the requirement for the award of degree of **Doctor of Philosophy (Ph.D.)** in the School of Computer Applications, is a research work carried out by Rajeev Kanday, 41800129, is bonafide record of his/her original work carried out under my supervision and that no part of thesis has been submitted for any other degree, diploma or equivalent course.

(Signature of Supervisor)

Name of supervisor: Dr.Ramandeep Singh

Designation: Professor

Department/school: School of Computer Science and Engineering

University: Lovely Professional University

ABSTRACT

In the rapidly evolving field of computing, Edge computing emerges as a ground breaking solution, offering real-time data processing capabilities directly at the data source, thus addressing the limitations associated with cloud computing, such as high latency, increased cost, and energy consumption. In this study, we have designed a smart framework that can identify the challenges with real time energy management solution in relation to edge computing-based health monitoring system. By leveraging edge computing, the system enables immediate, on-site analysis of vital patient data—blood pressure, heart rate variability, sugar level, oxygen saturation (SpO_2), positional data, in close proximity to the monitoring module—thereby significantly enhancing the speed and efficiency of medical response to various patient conditions, including fall detection. In this study, we have collected more than 40,000 patient scenarios from Guru Nanak Charitable Trust, Jalandhar (Pb), subsequently used to train a sophisticated classification algorithm. This algorithm integrates machine learning techniques to categorize patient states into three distinct conditions: no fall detected, slip detected, and definite fall, based on the analysis of six key physiological parameters. The effectiveness of the model was assessed through comprehensive data validation techniques, including box plots, histograms, correlation matrices, and rank graphs, to evaluate the predictive importance of each parameter. As per results it has been observed that the performance of random forest and decision tree algorithms are better than SVM classifier in terms of computational efficiency in association with energy efficiency, which also translates into substantial energy savings and its management. The implementation of a smart energy management framework further optimizes sensor node activation, extending battery life and ensuring sustainable operation throughout the modules. For energy optimisation, a frame work in relation to Energy Efficient Job Scheduling (E^2JS) has been developed and observed. From the results it has been estimated that the E^2JS saves approximately 30% energy consumption in comparison to regular task scheduling. For wireless transmission, the LoRA module was found to be the best alternate to WiFi and other wireless modules in terms of range, cost, and energy management. This research contributes to the field by demonstrating the practical application of edge computing in healthcare, where it can significantly reduce patient casualties by enabling prompt, accurate medical interventions. With an impressive accuracy rate of up to 95% in real-time patient condition prediction. The study not only showcases the potential of edge computing in

enhancing patient monitoring but also sets a benchmark for future research in the domain, particularly in the development of energy-efficient, real-time health monitoring systems.

Dedication

This thesis is dedicated to the loving memory of my mother, Mrs. Bimal Kanday, whose spirit and love continue to guide me through all of life's challenges. Her unwavering faith in me has been my beacon of light and source of strength, even in her absence.

Simultaneously, it is with immense gratitude that I dedicate this work to my father, Professor Ramesh Chand, whose support, encouragement, and unconditional love have been my pillars of resilience and perseverance. His presence and belief in my abilities have been instrumental in reaching this milestone.

To both my guiding stars, for their sacrifices, love, and blessings, this achievement is a tribute to you.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to all those who have made the completion of this thesis possible. First and foremost, I extend my sincere appreciation to my thesis advisor, Professor Ramandeep Singh, whose expertise, understanding, and patience, added considerably to my graduate experience. Your guidance and support from the preliminary to the concluding level enabled me to develop an understanding of the subject.

I am heartily thankful to my committee members, Dr. Manik Rakhra, Dr. Dalwinder Singh and Dr. Deepak Prashar whose encouragement, insightful comments, and hard questions pushed me to broaden my research from various perspectives. My sincere thanks also go to School of Compter Applications, School of Computer Science and Engineering and Division of Research and Development, Lovely Professional University, Punjab for providing the necessary resources and environment to complete my research.

I am also grateful to my colleagues Dr. Sachin Gupta, Mr. Jai Krishan, Ms. Tania Rajput and Mr. Arun for their camaraderie, stimulating discussions, and for all the fun we have had during the last few years.

Last but not least, I cannot forget to thank my family for their understanding, endless patience, and encouragement when it was most required. My Father, Professor Ramesh Chand, my spouse, Mrs. Lovina, my sons Mr. Reyansh Kanday & Mr. Harshit Kanday and my In – Laws have been my greatest supporters throughout this journey.

This thesis stands as a milestone in my academic journey, and these acknowledgments are a reflection of the teamwork and support behind this effort. To all of you, I express my profound gratitude and appreciation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1
Introduction

1.1 Introduction

In the "early days" of the Internet of Things (IoT), the processing and most of the storage were predominantly performed in cloud data centers because only the cloud had the computational resources needed to perform complex analysis. However, with an increasing deployment of connected applications in this field, the challenges and constraints of cloud processing also got magnified[1]. Consequently, the cloud has experienced a significant surge in workload, resulting in numerous implications that have been observed by the research community[2].

One of the major limitations was latency- the time required for the data generated by sensors at the source to traverse the path to the cloud for processing and then back to deliver actionable results. It is pertinent to note that the significance of latency varies in different systems[3]. On one hand, where milliseconds of might not be that important in a thermostat, on the other, it is highly imperative for industrial robots and other real-time systems, as they require very less time to operate, to guarantee safety and productivity[4]. For the sensor-based safety features on modern vehicles, latency can be a matter of life and death. It has been observed in the literature that the responsible applications can be executed with respect to high speed of data extraction.

As per the modern architecture of an IoT based scenario, even a modest application with sensors can create an enormous amount of data that consumes costly bandwidth that is provided by the network[5].

Today, security has become one of the most critical aspects of the IoT. The applications and users are seeking a very prominent level of security for the databases as the adoption of cloud-based approach may potentially expose sensitive information, including intellectual property (IP), that must be protected[6].

Based on the findings derived from the literature survey, it can be said that a better solution to overcome these challenges is to divide the processing tasks between the cloud-based servers and processors operating at data generation sites, commonly referred to as the edge[7]. More precisely, it is the edge of the network, or, from a data center's perspective, it is referred to as the 'far edge'. Note that some processing has always been performed at the edge, principally in gateways that aggregate the data produced by sensors into a standard format and then send it outward[8].

The Edge computing model is a paradigm that reimagines how computational resources are distributed within a networked environment, with the primary goal of reducing latency, improving real-time data processing, and enhancing the overall efficiency of applications[9]. In this model, computation and data processing are moved closer to the data source, which is often referred to as the "edge" of the network. This stands in contrast to traditional cloud computing,where data is sent to centralized data centers for processing. The Edge computing model is designed to address the limitations and challenges associated with centralized cloud architectures, particularly in scenarios involving data-intensive applications and emerging technologies like the Internet of Things (IoT) and real-time analytics[10], [11]. Here the rely on concepts related to edge computing have been discussed:

Key Concepts of the Edge Computing Model

1.1.1 Latency Reduction: One of the key driving factors for the adoption of the edge computing model is the reduction of latency. In the context of computing and data transmission, sometimes, there may be a delay between the client's request and the server's response[3], [12]. Especially in real-time applications like video streaming, online gaming, autonomous vehicles, and other Internet of Things (IoT) applications, high latency is totally undesirable as it can result in delay, lag, or poor user experience on the cloud. Here it is very important to understand the difference between traditional cloud computing and edge computing[13]. In a traditional cloud computing model, all datawhich was generated by devices, sensors, or users is sent to centralized data centers forprocessing and analysis. These data centers may be located far away from the source ofthe data, leading to increased latency due to the long round-trip time for data to travel[14]. Edge computing, on the other hand, places smaller, localized computing resources (likemicro data centers or edge servers) closer to the source of data generation. This can be within an IoT device itself or in a local gateway or even at a nearby facility[15]. By doing this, computing enables the data to be processed locally, thereby reducing the need for it to traverse extensive distances to centralized data centers[16]. Edge computing is one of the proven methods to reduce latency. Figure 1.1 shows the key aspects that are associated with an improvement in the latency factor of edge computing:

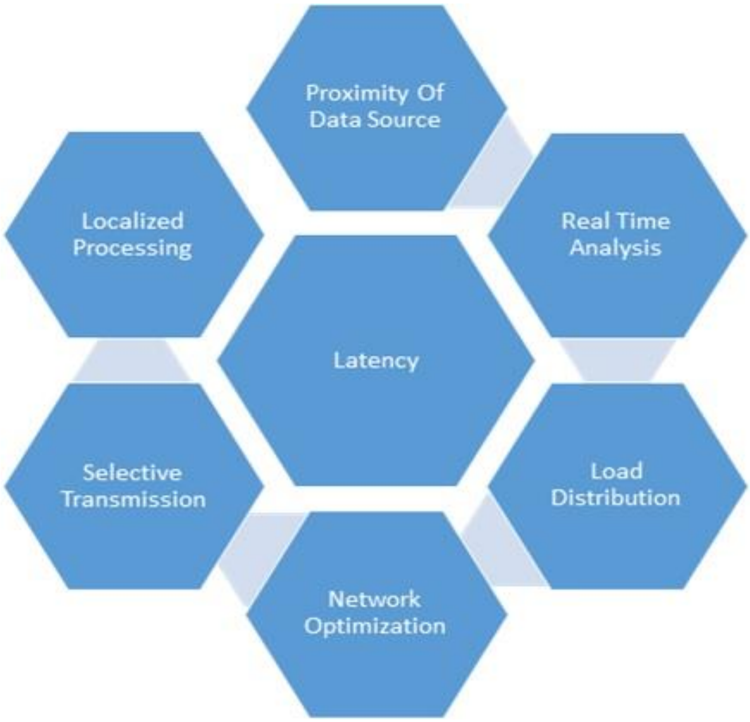

Figure 1.1: Key aspects in improvement of Latency rate in relation to edge computing

There are aspects by which latency has been improved, which are explained as below:

- ***Proximity to Data Source*:** There is a recognized need for data processing units to be situated in close proximity to the sources from which the data originates. By this method, the commutation time will be shortened, thereby resulting in a significant improvement in latency[17].

- ***Localized Processing*:** Edge computing allows for some data to be processed locally rather than being sent to a centralized data center as it is better to install the processingunits in the local territory rather than molding the data to cloud[18]. This localized processing can significantly enhance speed, thereby reducing latency up to a certain extent.

- ***Selective Transmission:*** It has been observed that the installation of smart data filters at the user end can improve the latency. Edge computing has the ability to filter and send only the necessary data to the centralized cloud, reducing the amount of data that needs to travel back and forth[19]. This saves bandwidth and reduces latency. For applications that require real-time decision-making, such as autonomous vehicles or industrial automation systems, edge computing allows for quicker data processing and decision-making[20].

- ***Network Optimization***- By performing computation at the edge, there is less strain on the central data centers, allowing them to perform more efficiently, which can also help reduce latency. The main benefit of edge computing methods is that it optimizes the network use with respect to the resources available in the IoT architecture[21].

- ***Load Distribution-*** When we talk about load distribution in edge computing, it refers to the equitable and even distribution of workload over all the available edge nodes[22]. This not only minimizes the total latency but also ensures that no single node becomesa bottleneck subsequently facilitating timely decision making[23].

- ***Real Time analysis-*** The presence of low latency expediate the processing and analysis of data, enabling systems to take immediate actions based on the findings derived fromthe examined data[24]. In systems such as autonomous vehicles or medical devices, where real-time responses are crucial for safety, reduced latency enables prompt data processing and execution of actions, thereby mitigating potential incidents[25].

Improving latency is one of the key attributes of edge computing thereby rendering edge computing a highly suitable solution for a diverse array of applications and industries that necessitate instantaneous data processing and decision-making capabilities[26].

1.1.2 Real-Time Processing: Real-time processing is another fundamental concept in edge computing that serves as a crucial enabler for applications requiring immediate or near-instantaneous data processing and feedback[27]. Unlike batch processing, which collects and processes data at predetermined intervals, real-time processing involves continuously handling of incoming data and delivering outputs with minimal delay[28]. The demand for real-time processing capabilities is driven by various applications and industries that require immediate analysis and action. This includes autonomous vehicles, health monitoring systems, industrial automation, and smart cities, among others [29]. Traditional cloud computing architectures are not always adequate for these types of applications due to the latency involved in transmitting data to and from a centralized data center. Real time processing can be easily formed in an edge computing architecture. The factors associated with real time processing are shown in figure 1.2.

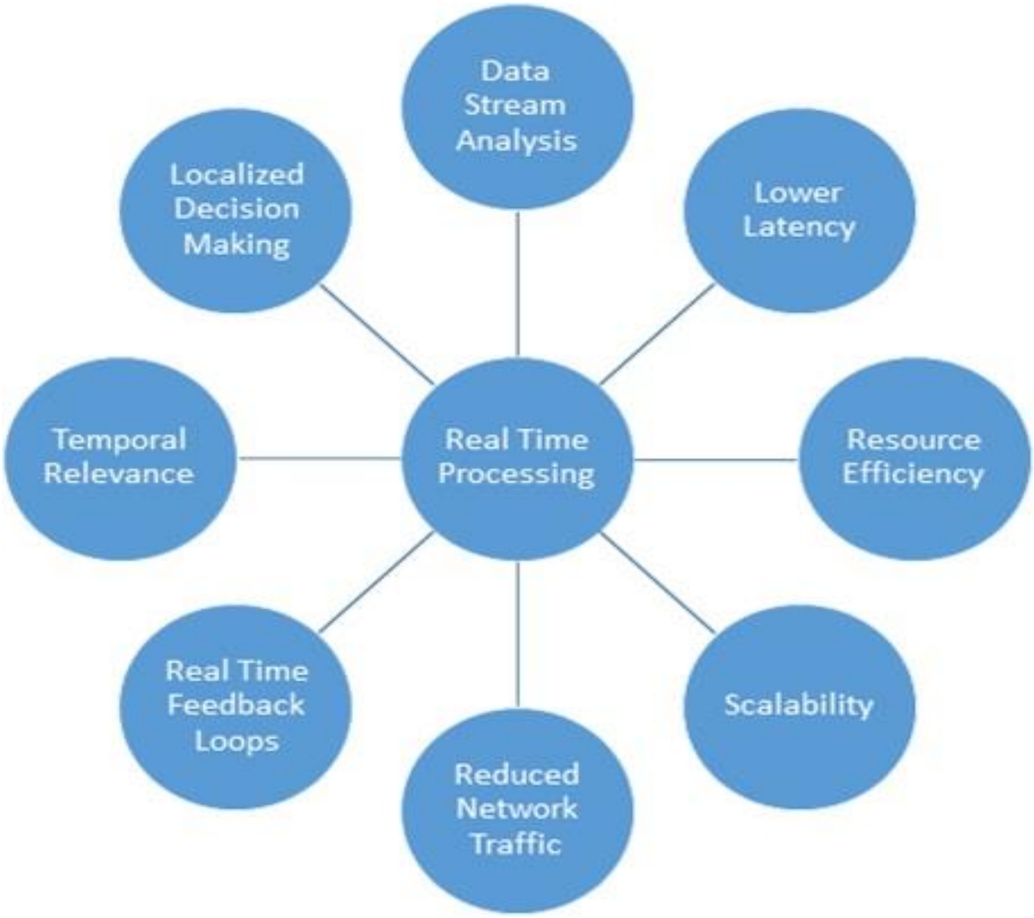

Figure 1.2: Factors associated with real time processing of data in edge computing.

- ***Localized Decision-Making:*** It has been observed in the literature that edge computing has much more optimized and proven power of computation than any other method in this problem statement[30]. Due to proximity of computing power to the data source, edge computing allows immediate analysis and prompt action, thereby eliminating the necessity of transmitting data to a central cloud for processing[31]. This is crucial for applications like autonomous vehicles, where even a slight delay can have significant consequences.

- ***Data Stream Analysis:*** Edge computing can analyze data streams in real-time, thereby facilitating the implementation of applications such as real-time video analytics, real-time analysis of sensor data in industrial environments, and real-time monitoring of financial transactions[32]. The proximity of data sources reduces the time taken for data to travel, enabling quicker decision-making and implementation. Additionally, the localization of data processing at the edge not only minimizes the volume of data that needs to be transmitted over the network, but also reduces network congestion and latency[33].

- ***Temporal Relevance:*** Some data loses its relevance quickly. For example, sensor datain a manufacturing process may only be relevant at that moment and loses its importance over time[34]. To overcome this challenge, edge computing ensures prompt execution of time-critical tasks based on the most up-to-date information available and enables real-time data processing. Edge computing can perform data filtering, aggregation, and pre-processing, ensuring that only necessary data is sent to the cloud[35]. This makes efficient use of bandwidth and storage resources, which is essential for real-time applications that generates a large volume of data.

- ***Real time feedback loops-*** The real-time feedback loops allows the applications to require continuous monitoring and adjustment, such as process control systems in manufacturing or energy grids[36]. These applications can derive advantages from the real-time feedback loops made possible by edge computing.

- ***Reduced Network traffic-*** One of the important features of edge computing is that it enables the analysis of data in close proximity to its source[37]. Because of this, the necessity of transmitting the data to a centralized point gets reduced. Real-time processing in edge computing, hence, significantly reduces network traffic[38].

- ***Scalability-*** Edge computing allows for decentralized processing, which makes it easier to scale real-time applications[39]. As new edge nodes can be added with localized computational resources, real-time processing capabilities can be scaled horizontally without overloading a central server.

- ***Resource Efficiency-*** The localization of data processing decreases the amount of data that needs to be transferred to a centralized source, resulting in a reduction of energy consumption and resource utilization associated with data transmission across networks along with the associated costs[40].

- ***Lower Latency-*** Real-time processing in edge computing is inherently meant to reduce latency, which pertains to the time delay between a system's response to input or stimulus[41]. Instead of sending data to a centralized data center for analysis and processing, edge computing enables data to be processed locally, further reducing delay[13]. This is significantly important for those applications that necessitate immediate and instantaneous feedback, such as, industrial automation, and various Internet of Things (IoT) applications[14].

So, the real-time processing is integral to the edge computing paradigm and serves as a key enabler for various applications that require immediate or near-instantaneous decision-making and action[15]. By moving computation closer to data sources and enabling real-time analysis and feedback, edge computing opens the door to new possibilities in IoT, industrial automation, healthcare, transportation, and many other sectors.

1.1.3 Bandwidth Optimization: Bandwidth optimization is another critical aspect of edge computing that addresses the limitations and costs associated with data transmission over networks[42]. Bandwidth refers to the maximum rate of data transfer across a network path and optimizes both performance and cost-effectiveness. One of the primary objectives of a smart edge computing approach is to use the bandwidth in an effective way[43]. Transmitting large volumes of data to centralized cloud data centers can strain network bandwidth and lead to congestion. Edge computing decreases the necessity of transmitting data over a network by performing processing and analysis near the data source[44]. This effectively optimizes bandwidth usage and mitigates network congestion. Figure 1.3 shows the bandwidth management factors in relation to edge computing methods:

- ***Local Processing***- As IoT and other data-intensive applications grow, so does the volume of data generate. And transmitting this data over long distances to centralized data centers can be expensive, especially for organizations that pay based on the amount of data they transmit. Efficient use of bandwidth is crucial for these applications to scale without overwhelming the network[44]. The storage of locally processed and filtered data can effectively minimize the requirement for massive data transfers and improve the management of data at the local level[43].

Figure 1.3: Factors associated with Bandwidth management in edge computing

- ***Load balancing***- In order to avoid network congestion and optimize bandwidth, Load balancing evenly distribute workloads across multiple computer resources. It also assures smooth and uninterrupted operations[45]. Bandwidth optimizing aids in rapid transfer of data between nodes. It plays a crucial role in maintaining optimal performance by minimizing latency and enhancing the speed of data retrieval and processing[46].

- ***Adaptive streaming***- Bandwidth optimization enables adaptive streaming algorithms to dynamically modify the streaming content's quality in response to the prevailing bandwidth conditions[47]. This ensures a smooth streaming experience with little buffering and streams high-quality content whenever sufficient bandwidth is accessible. Inadequate bandwidth, on the other hand, can degrade the quality of services like video streaming, online gaming, and real-time analytics, leading to poor user experience[43].

- ***Protocol Optimization***- Edge devices can use more efficient data transmission protocols that are tailored for local network conditions, further optimizing bandwidth usage[48]. In multimedia applications, edge computing can dynamically adjust the quality of video or audio streams based on current network conditions. The main reason of edge computing technology is to distribute workload across multiple edge servers which can reduce bottlenecks and make more efficient use of network resources[49]. With real-time analytics, edge computing can adapt to network conditions dynamically, ensuring that bandwidth is utilized as efficiently as possible[50].

- ***Caching-*** Caching plays a pivotal role in optimizing bandwidth as it effectively reduces the volume of data that needs to be transmitted between the edge device and the cloud[51]. As a result of which, the overall data traffic that traverses the network decreases. By locally delivering requests using cached data, the network's overall burden is diminished, resulting in reduced congestion and optimal utilization of the available bandwidth[52]. This is particularly useful for content delivery networks (CDNs) to improve user experience while conserving bandwidth[53].

- ***Data Aggregation and filtering-*** By implementing edge computing, a significant portion of the data can be analyzed without the need for transmission to a central server[35]. Edge computing devices can aggregate raw data and filter out irrelevant information before sending it to the central cloud, which reduces the volume of data that needs to be transmitted over the network.

- ***Off-peak data transfer-*** In edge computing methodologies, it is possible to strategically schedule data transfers during periods of non-real-time workloads, such as off-peak hours[54]. This approach allows for the efficient utilization of network resources, hence optimizing their usage.

Edge computing allows for local peer-to-peer data sharing, eliminating the need to transmit data over long distances. By optimizing bandwidth usage, edge computing not only improves application performance and user experience but also helps in reducing the operational costs[55]. This makes it a key enabler for applications that generate large volumes of data, require low latency, or need to operate in bandwidth-constrained environments.

1.1.4 Privacy and Security: Privacy and security are critical aspects that must be considered in any computing architecture, including edge computing[56]. Applications, especially those involving sensitive data, are getting benefitted from edge computing's ability to process data locally, which in turn enhances data privacy and security by minimizing the exposure of sensitive information to potential threats during transit[57]. While edge computing offers several advantages in terms of latency reduction, real-time processing, and bandwidth optimization, it also introduces new challenges and opportunities for privacy and security[58]. Figure 1.4 shows the strategies for enhancing privacy and security in edge computing architecture:

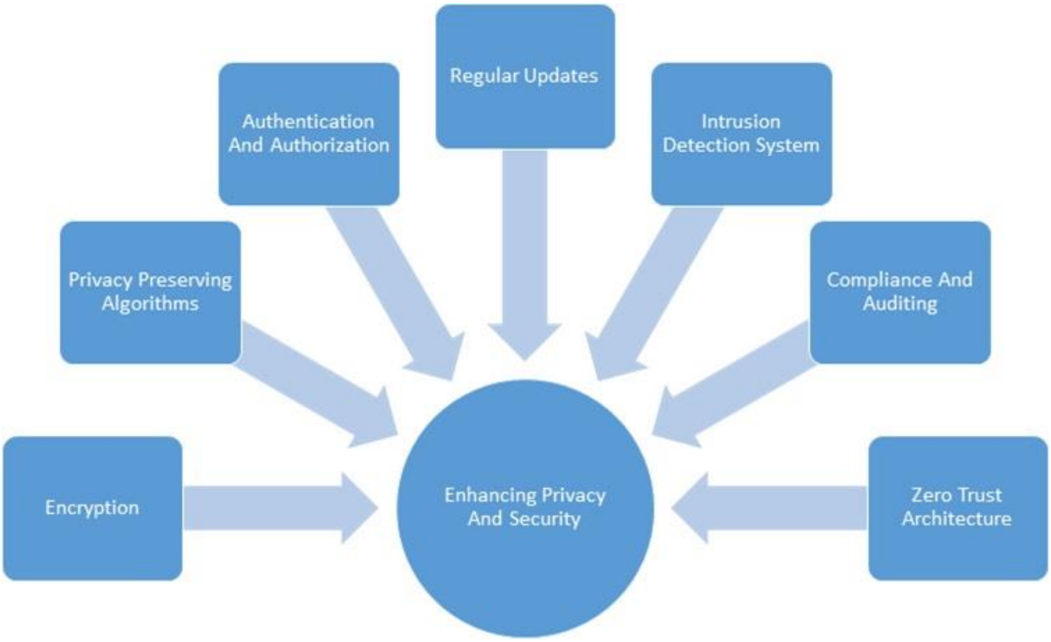

Figure 1.4: Strategies for Enhancing Privacy and Security in Edge Computing

- ***Encryption-*** Just like real-time data processing, edge computing can also enable real-time security monitoring, allowing for quicker detection of and response to security incidents. Data can be encrypted both at rest and in transit to safeguard against unauthorized access[59]. Robust mechanisms can be implemented to ensure that only authorized devices and users can access the edge nodes and the data stored or processed on them[60].

- ***Zero trust architecture (ZTA)-*** The concept of Zero Trust Architecture (ZTA) revolves around the principle of "never trust, always verify." This implies that trust is not automatically assumed, instead, verification process is obligatory for any organization seeking access to network resources, irrespective of their location[61]. Adopting a zero- trust approach, which assumes that no devices within the network are secure, can provide an additional layer of security. This concept is significantly important in edge computing, wherein devices, data, and services are distributed across different geographical locations and network environments, potentially exposing them to various vulnerabilities[62].

- ***Compliance and auditing-*** In sectors such as healthcare and finance, which often operate under stringent laws pertaining to the storage of data, it is important for edge computing to align with the data sovereignty regulations which mandate that data must be stored and processed within specific geographic limits[63]. With capabilities to process data locally, only necessary information may be sent to the central cloud. This selective transmission can reduce the amount of sensitive or personal data from getting exposed to potential risks during data transfer[64]. Edge computing allows for more granular control over what data is sent back to central servers, potentially providing users with more control over their personal data.

- ***Intrusion detection system (IDS***)- The decentralized nature of edge computing can expand the attack surface. Each edge node becomes a potential entry point for attackers, making the overall system more vulnerable to attacks like data tampering or unauthorized access[65]. Localized processing means that data might not always be sent to a central server. Keeping the software up to date is crucial for mitigating these known vulnerabilities. This is particularly challenging but essential for edge devices. Employing IDS (Intrusion detection system) on edge devices can monitor and alert potentially malicious activity, allowing for quicker remedial actions. This approach can be advantageous as far as privacy is concerned, but at the same time, it can also present problems in terms of upholding a unified and authoritative information repository, consequently leading to data integrity issues[66]. However, to some extent, Edge nodes can be isolated from each other and from the central system, which can contain attacks or vulnerabilities to a smaller part of the network.

- ***Regular updates***- Regular updates in edge computing plays a crucial role in maintaining a secure environment. These updates serve to address vulnerabilities, improve, and optimize security features, protect data privacy, and equip the system with the ability to counter emerging threats[56]. The implementation of a structured, reliable, and uniform updating approach is crucial in maintaining privacy and security of edge computing deployments.

- ***Authentication and authorization-*** Authentication and authorization plays a very important role in ensuring privacy and security in edge computing. Authentication is the process of verifying the legitimacy of a person, device, or service in order to establish their true identity[67]. Typically, the process of authentication is accomplished by employing usernames, passwords, tokens, biometrics, or other forms of verification. The implementation of robust authentication measures is crucial in order to restrict data access solely to authorized entities, thereby mitigating the risk of unauthorized access that may result in data breaches[68]. Once authentication of an entity is done, authorization comes into picture which involves the determination of the activities or resources that the authenticated entity is permitted to access or alter. This determination is made based on specified policies. In short, both authentication and authorization function as gatekeepers that govern the access to data and services within a network, subsequently exerting a direct influence on privacy and security[69].

- ***Privacy preserving algorithms***- For cases like data analytics, employing algorithms that can process data without revealing sensitive information can help in maintaining user privacy[68]. By taking a comprehensive approach to privacy and security, edge computing can be made both robust and resilient, offering a viable solution for various applications that demand low latency, real-time processing, and bandwidth optimization, all while keeping data secure and private[58]. Edge devices may be constrained in terms of computational resources, limiting the types of security measures that can be implemented directly on the devices. Ensuring that edge computing architectures meet relevant industry regulations and standards is crucial for both privacy and security[70].

1.1.5 Scalability: Scalability is a critical factor for the long-term viability and success of any computing architecture, which includes edge computing as well. Edge computing supports a distributed architecture that can be easily scaled by adding more edge nodes or servers as needed[39], [71]. Figure 1.5 shows the modular approach in improvement of scalability in relation to Edge computing.

- ***Network Efficiency***- Edge computing offers flexibility that allows the network to adapt to changing demands without relying solely on centralized data centers[72]. As applications and systems grow, it becomes imperative for the underlying infrastructure to effectively accommodate this expansion. This may involve managing larger volumes of data, enhancing data processing speed, or supporting a greater number of users[73].

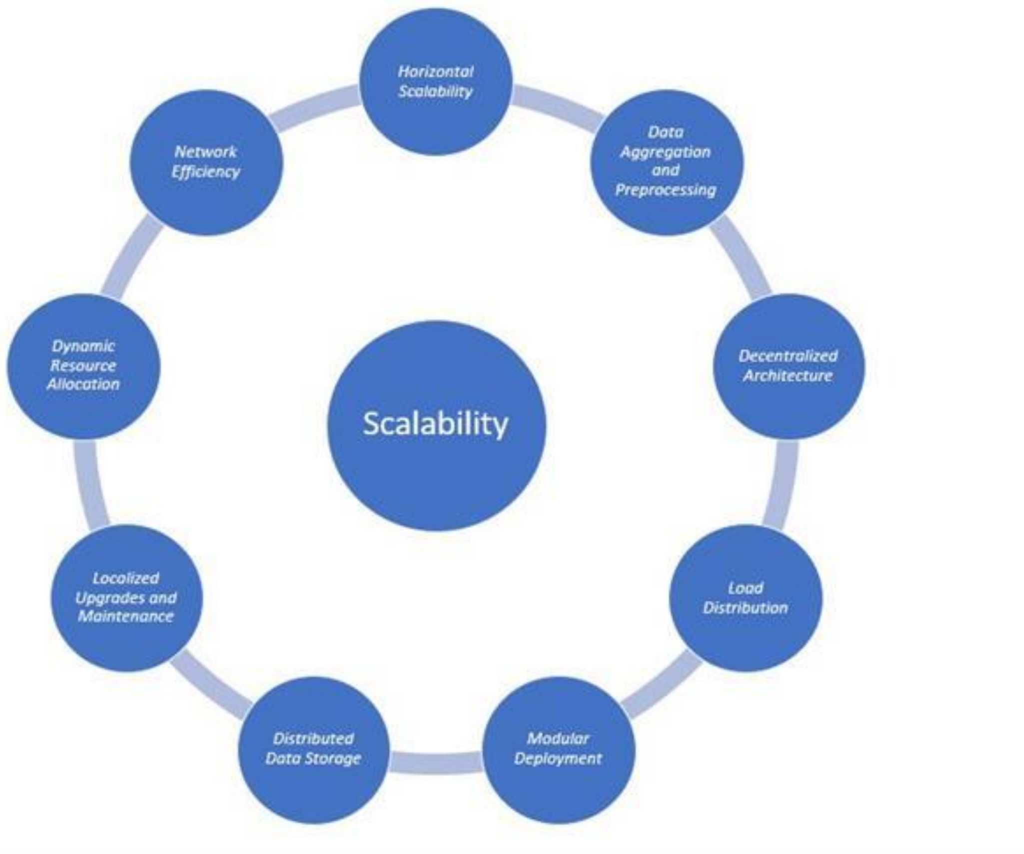

Figure 1.5: Key aspects to improve the scalability in Edge computing architecture.

- ***Dynamic Resource Allocation-*** New applications are continually emerging that require real-time processing, machine learning, and complex analytics. A scalable architecture is needed to meet these varying computational demands[74]. As applications grow, maintaining a high level of service is crucial. Scalability ensures that performance doesn't degrade as the user base expands. A scalable architecture allows resources to be allocated more efficiently, which can result in cost savings over time[75].

- ***Localized Upgrades and Maintenance-*** As networks expand, it becomes highly imperative to upgrade or substitute certain nodes in order to support emerging technologies or meet new and ever-changing demands. Here, localized maintenance fulfills this requirement by providing the ability to upgrade, repair, or replace certain edge nodes or devices without causing any disruptions to the overall network functionality. It also ensures data consistency and dependability among several nodes. Not only this, it also limits the consequences of failure, if any, to a specific node or portion[76].

- ***Distributed data storage-*** Edge computing is a distributed computing paradigm that involves the storage of data at several edge nodes, resulting in enhanced efficiency and faster data retrieval as the demands of the system increase[77]. In an edge computing environment, individual nodes can often be upgraded or maintained without affecting the entire network, thereby offering another layer of flexibility in scaling the system. Advanced edge computing solutions can allocate resources dynamically based on the current computational and storage demands, thus providing highly efficient, scalable operations[78].

- ***Modular Deployment-*** Edge computing allows for modular deployments where additional resources can be deployed only where and when they are needed. This simplifies the process of scaling operations in a manner that is both efficient and economical[79].

- ***Load distribution-*** The implementation of an effective load distribution system among edge nodes prevents the occurrence of node overload, hence assure stability and optimal performance of activities[80]. Also, the distribution of workloads facilitates the localization of data processing in close proximity to data sources, resulting in decreased latency and improved user experience[45].

- ***Decentralized Architecture-*** The decentralized nature of edge computing means that new edge nodes (devices, servers, etc.) can be added to the network as needed, without causing a bottleneck at a central server[81]. Distributing computing tasks across multiple edge locations can balance the load, reducing the risk of failure at any single point and enabling more efficient resource utilization[77].

- ***Data Aggregation and Pre-processing-*** Edge devices can aggregate and preprocess data locally. This not only conserves network bandwidth but also enables the transmission of more significant and condensed information to centralized data centers, consequently, optimizing the utilization of resources within these data centers[35].

- ***Horizontal scalability-*** Edge computing offers several advantages and strategies to improve scalability, which is particularly crucial for IoT, real-time analytics, and other emerging technologies that are generating unprecedented volumes of data. With billions of IoT devices and sensors deployed worldwide, the volume of data being generated is staggering. Scalability is crucial to handle this massive influx of data efficiently[71]. Unlike traditional centralized systems that often rely on scaling up (adding more power to a single machine), edge computing naturally lends itself to horizontal scaling—adding more machines into the network. This is especially effective for handling increased data volume and computational load[82].

The concept of scalability plays a crucial role in edge computing. It allows the architecture to successfully adapt to the increasing and unpredictable demands effectively. By minimizing the distance that data needs to travel, edge computing makes the better use of network resources, thus supporting scalability in terms of both data transmission and processing speed[39]. By leveraging edge computing's innate ability to scale horizontally, distribute load, and dynamically allocate resources, organizations can build robust, efficient, and future-proof systems[83].

1.1.6 Offline Operation: Offline operation is a significant advantage of edge computing, particularly for applications and environments where continuous connectivity to a central cloud or data center is not guaranteed. Systems that can operate offline are less susceptible to network outages and can continue functioning even when central services are unavailable, thereby enhancing the system's overall resilience[84]. Offline operation can also reduce the costs associated with data transmission, especially over cellular or other metered networks[85]. The ability to function without an internet connection is advantageous especially for applications that necessitate real-time data processing, since it mitigates the latency that would otherwise arise from transmitting data to and from a central server. Offline operation at the edge is also beneficial for ensuring compliance with data governance standards that mandate the confinement of data within specific geographic borders[12]. In consumer applications, it ensures that users have access to essential features even when they are temporarily disconnected from the internet. Figure 1.6 shows the association of offline operations with edge computing methods for improvement in services:

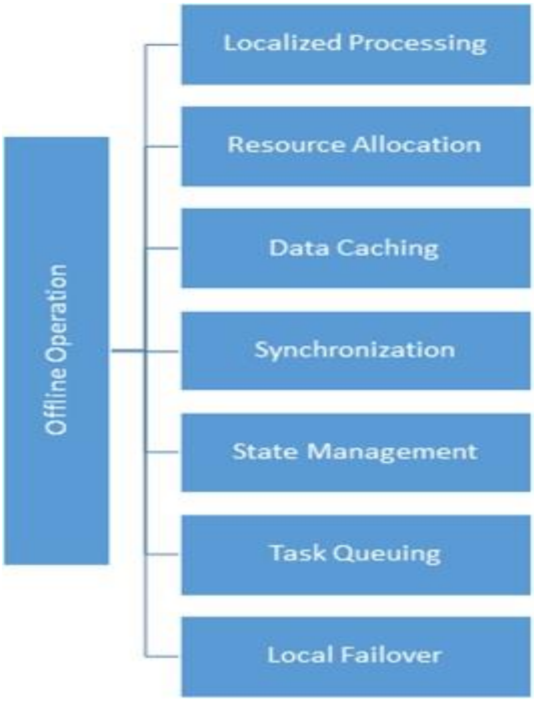

Figure 1.6: Association of offline operations with edge computing methods

- ***Local failover***- Many edge computing applications are in remote or challenging environments like offshore oil rigs, rural agricultural settings, or isolated industrial sites where connectivity can be sporadic. Unlike traditional cloud computing models, that rely on uninterrupted internet access for data processing and storage, edge computing allows for localized processing and decision-making, enabling operations to continue even when a device is disconnected from the network[10].

- ***Task Queuing***- Operations that require central server interaction can be queued for execution once connectivity is restored, thereby preventing data loss or operational hiccups. Even in the absence of an internet connection, devices can gather and retain data in a local storage system. In more complex edge architectures, if one edge device fails or loses connectivity, other nearby edge devices can take over its tasks, providing a level of redundancy[32]. After getting reconnected to the internet, the queued tasks are executed, and the data can be synced with central data centers or cloud resources, thereby ensuring the maintenance of data consistency. The ability to queue tasks allows the system to effectively manage variations in network availability without compromising operational functionality[33]. Systems can adjust dynamically when the network conditions are poor and subsequently synchronize when the connectivity re- establishes.

- ***State Management-*** In edge computing, the term "state" pertains to the stored data that encompasses the information characterizing the condition or state of a system or process at a specific moment in time. Advanced edge computing solutions can manage the state of an application locally, ensuring that it remains consistent even when transitioning between online and offline operations[29]. An effective state management guarantees that when a device regains the lost internet connectivity, the process of data synchronization takes place in such a manner that maintains data consistency and integrity, assuring a seamless user experience.

- ***Synchronization-*** After getting re-connected with the connection, edge devices have the ability to synchronize with central servers, facilitating the transfer of newly acquired data to the central servers, along with the retrieval of updates or new settings from the central servers to the edge devices.

- ***Data Caching-*** Edge devices can cache essential data locally, providing the necessary information for operations to continue even when the device is disconnected. The concept of data caching in edge computing refers to the practice of storing data in a cache, which is basically a layer of high-speed data storage. This approach aims to minimize latency and optimize bandwidth use when fulfilling data requests[12]. The utilization of cached data guarantees the continued accessibility of essential information to users, even in situations where online connectivity is unavailable, contributing to a smooth and uninterrupted user experience.

- ***Resource Allocation-*** Resource allocation is a process that involves the prioritization and scheduling of tasks to effectively maintain the smooth operation of important functions during offline operations. Not only this, effective allocation of resources also assures the maintenance of operational continuity, data integrity, and enhanced user experience[3]. It also helps in maintenance of fault tolerance along with the efficient management of network communications following its reconnection.

- ***Localized Processing-*** Since, data is not sent to the centralized data centers, Localized processing enables edge devices to operate autonomously without any dependence on cloud connectivity. Even if the connection to a central server is lost, the edge device retains the capability to process data and provide services independently[18]. The optimization of data processing and local accessibility facilitates offline operations by supplying essential data inputs for local applications and services.

Edge devices can be equipped with computational power, storage, and memory to operate independently, ensuring they have the resources to continue functioning offline. Offline operation is particularly valuable for applications that require high availability, low latency, or operation in remote or challenging environments[10]. By enabling devices to act intelligently on their own, edge computing creates more robust, resilient systems that can continue to operate effectively, even when disconnected from the central cloud or data center.

Components of the Edge computing model

The edge computing has basically four components which are given below:

1. **Edge Nodes:** These are the devices or servers located at the edge of the network, such as routers, switches, gateways, and IoT devices. Edge nodes are equipped with computational capabilities and storage resources, enabling them to perform processing tasks and store data locally.

2. **Fog Computing:** This term is sometimes used interchangeably with edge computing. It refers to a network architecture where computational tasks are offloaded from end devices to intermediate nodes, like routers and switches[86]. These intermediate nodes, known as "fog nodes," provide additional processing power and storage capacity.

3. **Mobile Edge Computing (MEC):** MEC specifically focuses on leveraging resources available at cellular base stations to offload computation from mobile devices. This approach enhances mobile application performance by reducing latency and optimizing data processing[87].

4. **Edge Servers**: These are dedicated computing nodes placed strategically at the edge of the network. They provide more substantial processing power and storage compared to edge nodes, making them suitable for handling more complex tasks and applications[11]. The architecture of edge computing is shown in figure 1.7.

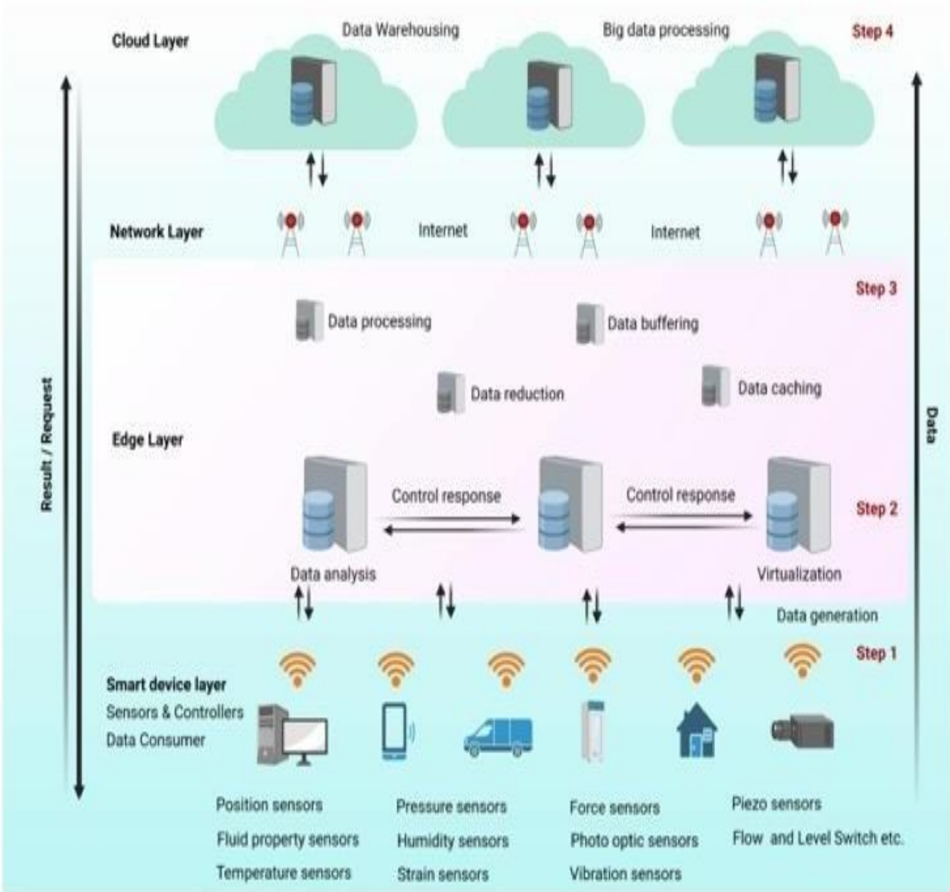

Figure 1.7: Edge computing architecture

1.2 Emerging Applications Requirement

Emerging applications, driven by technological advancements and changing user demands, often require specific capabilities that traditional computing paradigms struggle to fulfill. Edge computing has emerged as a solution to meet the unique requirements of these applications, offering advantages in terms of reduced latency, real-time processing, and efficient data management[88]. As shown in figure 1.8, there are of various emerging applications:

1.2.1 Augmented Reality (AR) and Virtual Reality (VR)

It has been observed that the AR and VR applications demand ultra-low latency to provide seamless and immersive user experiences. Delays in rendering virtual objects or environments can lead to motion sickness and break the illusion of immersion. Edge computing is a technological approach that aims to minimize latency by executing rendering and interaction duties in proximity to the user's device. This minimizes the round-trip time between the user's device and a remote cloud server, guaranteeing prompt delivery and real-time updating of virtual content. This is critical to maintaining the fluidity and realism of AR and VR experiences[86].

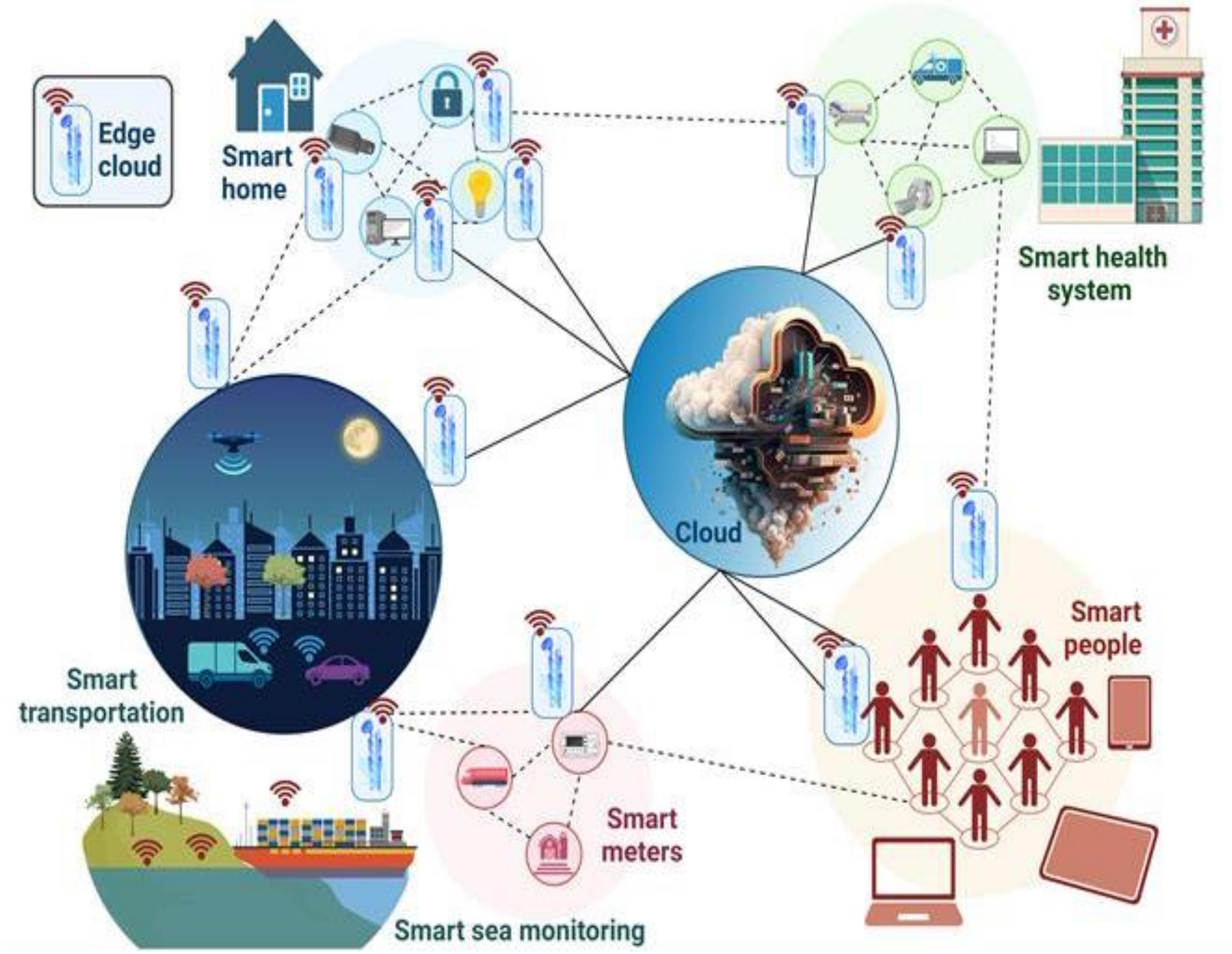

Figure 1.8. Emerging applications of Edge Computing

1.2.2 Internet of Things (IoT) and Smart devices

IoT devices generate substantial amounts of data, and many of them require real-time responses for efficient control and automation. The transmission of all IoT data to a centralized cloud can enhance latency and result in the formation of bottlenecks and elevate the cost. Edge computing allows IoT data to be processed and analyzed at the edge nodes, reducing the need to transmit all data to the cloud. Local processing also supports immediate actions, such as adjusting equipment settings or triggering alerts, based on real-time data. This technique not only mitigates latency but also facilitates data filtration and prioritization, selectively transmitting pertinent information to the cloud for subsequent analysis.

1.2.3 Autonomous Vehicles

Autonomous vehicles require split-second decision-making based on sensor data, such as lidar, radar, and camera feeds. Delays in processing can even have life-threatening consequences in critical situations. Edge computing enables on-board processing of sensor data, allowing vehicles to make instantaneous decisions without relying on distant cloud servers[89]. This rapid response time not only reduces the latency but also enhances the safety within autonomous driving systems.

1.2.4 Smart Transportation

Smart transportation systems enhance traffic management, improve infrastructure use, provide better transportation services, and also increase users' traffic safety and efficiency. To optimize traffic flow, energy consumption, and public services, real-time data processing is essential. Immediate processing of traffic data allows the system to identify and respond to traffic congestion promptly, potentially rerouting vehicles to avoid heavy traffic. It also provides smart parking solutions. Not only this, by processing data from various sensors[90], it helps to predict when parts of vehicles or infrastructure require maintenance.

1.2.5 Healthcare and remote monitoring

Telemedicine applications and remote patient monitoring require real-time analysis of patient data to provide timely medical interventions and diagnostics. Edge computing allows wearable medical devices to process and analyze patient data locally. This approach mitigates the necessity of transmitting sensitive health data to remote servers, thereby augmenting patient confidentiality and facilitating expedited medical interventions[63]. At present, robotic manipulator-based surgery from remote locations is the prime demand of edge computing architecture. The demand of edge computing-based application is very high in the field of telemedicine primarily driven by the need to replicate the competence of an expert in a remote place for the purpose of treating patients[64].

1.2.6 Industrial Automation and Industry 4.0

With the evolution of industry from Industry 1.0 to Industry 4.0, the need for rapid response time in industrial automation applications has also increased, for process control, quality assurance and predictive maintenance. Edge computing supports real-time data analysis within manufacturing environments. Sensors can collect data on machine performance and product quality, thereby enabling instant adjustments and preventing costly downtimes[91]. The server communication establishment on edge computing model paves the way to new era in the industrial advancements. In future, the involvement of Edge computing may begin the era of industry 5.0[92].

1.2.7 Smart Homes

Edge computing allows for the processing of data locally, ensuring quick responses from smart home devices, which is crucial for functionalities like security systems and timely dissemination of emergency alerts. Edge computing enables devices to communicate with each other in real-time, enhancing automation scenarios like lighting adjustments, heating, or cooling based on occupancy data.

1.2.8 Smart Sea Monitoring

Edge computing plays a crucial role in making instant navigational decision-making by processing data from sensors onboard. It allows autonomous vessels to swiftly respond to obstacles, ensuring safe navigation. Swift processing of meteorological data allows for rapid broadcasts and alerts related to storms, high waves, or other hazardous conditions. It can also track marine life and biodiversity in real-time by processing data from sensors and cameras[93]. Furthermore, Edge computing can optimize shipping routes based on current sea conditions, traffic, and other relevant factors.

1.3 Edge Computing Approaches

Edge computing encompasses various architectural approaches that distribute computational resources and data processing closer to the edge of the network. These approaches address the challenges posed by latency, bandwidth constraints, and real-time processing requirements in different ways. Here, two primary edge computing approaches are discussed: Fog Computing and Mobile Edge Computing (MEC).

1.3.1 Fog Computing

Fog computing extends the cloud computing paradigm to the edge of the network. It leverages intermediate nodes, such as routers, switches, and gateways, as fog nodes to provide computational resources and storage capabilities. Fog nodes are strategically placed between end devices and centralized cloud servers, forming a hierarchical architecture. Fog nodes process data locally, reducing the need to transmit all the data to distant cloud data centers[86]. They can perform tasks such as data filtering, aggregation, and pre-processing, as well as hosting applications closer to the data source.

Benefits:

- **Reduced Latency:** By processing data closer to the source, fog computing minimizes communication delays and improves response times.
- **Bandwidth Optimization:** Only relevant or summarized data is sent to the cloud, optimizing network bandwidth, and reducing congestion.
- **Scalability:** Fog nodes can be easily added or removed to adapt to changing network demands.
- **Real-Time Analysis:** Fog nodes support real-time data analytics and decision-making, enhancing the capabilities of applications that require immediate responses.

1.3.2 Mobile Edge Computing (MEC)

Mobile Edge Computing (MEC) focuses on utilizing resources available at cellular base stations or access points to offload computation from mobile devices. It is particularly relevant in mobile networks and addresses latency concerns for mobile applications. MEC enables mobile devices to offload processing tasks to nearby edge servers located at base stations. These edge servers provide computational power and storage for applications running on mobile devices[87].

Benefits:

- **Latency Reduction:** By processing data locally at base stations, MEC minimizes latency for mobile applications.
- **Improved Mobile Application Performance:** MEC enables resource-intensive tasks, like video processing and real-time analytics, to be offloaded from mobile devices, improving their performance.
- **Network Efficiency:** Offloading computation to edge servers reduces the load on the core network and improves overall network efficiency.
- **Location-Aware Services:** MEC leverages location information to provide context-aware services to mobile users.

1.3.3 Hybrid Approach

There is no one-size-fits-all solution in edge computing, and hybrid approach, combining elements of both the fog computing and MEC are also explored. This approach seeks to optimize edge resources based on specific requirements of applications and network architectures[39]. While edge computing offers benefits, it also presents challenges such as managing heterogeneity among edge devices, orchestrating tasks across distributed nodes, ensuring data consistency, and maintaining security and privacy in a decentralized environment. Addressing these challenges requires careful architectural design, efficient resource allocation, and robust communication protocols[81].

1.4 Edge Computing in Health care

Edge computing has revolutionized various industries, and healthcare is no exception. As per reports by scientific groups, edge computing plays a pivotal role in health care and patient treatment processes. There are various crucial stages in the health care model where there is a need for a fast and trusted communication architecture for the care of patients in an emergency situation[63]. In the healthcare sector, where timely access to data and real-time decision-making are crucial, edge computing offers transformative solutions. From remote patient monitoring to personalized treatment plans, edge computing enhances patient care, improves operational efficiency, and addresses critical challenges faced by the healthcare industry[64]. The most popular domains are mentioned below:

1.4.1 Remote Patient Monitoring

Remote patient monitoring is crucial for healthcare practitioners, especially in cases of chronic illness or post-operative care, while minimizing delays in data collection and analysis. Edge devices, such as wearable health trackers and smart medical devices, can process and analyze patient data locally[58]. This reduces the need to send large amounts of sensitive health information to remote servers, improving patient privacy and minimizing latency. Real-time analysis also enables immediate intervention if a patient's vital signs indicate an emergency[57].

1.4.2 Real-Time Diagnostics

Rapid and accurate diagnosis is essential in critical healthcare scenarios. Edge devices equipped with advanced sensors and machine learning algorithms can analyze medical images, such as X-rays or MRI scans, locally. This will speed up the diagnostic process by providing preliminary assessments in real-time. This enables the healthcare professionals to take quicker decisions, especially in emergency situations[34].

1.4.3 Personalized Treatment

Real-time processing and prompt analysis of data play a crucial role in situations where treatments are to be customized to suit the specific requirements of individual patients. Edge devices can aggregate and analyze patient data, including medical history, genetics, and real-time monitoring data[63]. The data obtained is subsequently utilized to create personalized treatment strategies, thereby enhancing the overall outcomes for patients. Also, the implementation of local processing within healthcare facilities ensures the privacy and security of sensitive patient data by confining it within the premises. Handheld and non-invasive devices are adopting the edge computing facility for continuous monitoring of biological signals at a remote location[64].

1.4.4 Telemedicine and Remote Consultations

Telemedicine consultations require seamless video conferencing and real-time data sharing between patients and healthcare providers. Edge servers deployed at healthcare facilities can facilitate telemedicine consultations by ensuring high-quality video streaming, low-latency communication, and secure data transmission[94]. Furthermore, edge devices located at the residences of patients have the capability to gather and transmit data while engaging in virtual consultations, thereby furnishing healthcare professionals with precise and up-to-date information.

1.4.5 Emergency Response

In emergency situations, immediate access to patient information and real-time communication among healthcare professionals is of utmost importance. Edge devices can provide instant access to patient records, allergies, medications, and medical history during emergencies. Edge-enabled communication tools facilitate rapid collaboration between healthcare teams, ensuring quick decisions and responses[58].

1.4.6 Predictive Maintenance of Medical Equipment

Medical equipment maintenance is essential to prevent operational disruptions and ensure patient safety. Edge-enabled sensors installed on medical devices can monitor their performance in real-time. This data is processed locally to predict potential malfunctions, enabling timely maintenance, and reducing the risk of equipment failures during critical procedures[57].

1.4.7 Data Privacy and Compliance

Healthcare data is highly sensitive and subject to strict privacy regulations. The localization of sensitive patient data on edge devices mitigates the necessity of transmitting such data to distant servers, thereby augmenting the level of data privacy[95]. Edge computing also enables healthcare facilities to maintain compliance with regulations like HIPAA (Health Insurance Portability and Accountability Act) by keeping patient information within controlled environments.

1.5 Edge Computing Service

Edge computing services encompass a range of functionalities and offerings that enable efficient deployment, management, and optimization of computational tasks, data storage, and applications at the edge of the network. These services play a crucial role in leveraging the benefits of edge computing while addressing the challenges of latency, bandwidth constraints, and real-time processing requirements. The key aspects of edge computing services are given below:

1.5.1 Task Deployment

Task deployment involves distributing computational tasks to appropriate edge nodes based on factors such as task requirements, resource availability, and network conditions.

Benefits:

- **Latency Reduction:** Task deployment ensures that tasks are executed closer to the data source, minimizing latency.
- **Resource Efficiency:** Tasks are allocated to edge nodes with available resources, optimizing resource utilization.
- **Load Balancing:** Task deployment services balance the computational load across edge nodes, preventing resource bottlenecks.

1.5.2. Storage

Edge storage services involve managing and optimizing data storage at edge nodes[96].

Benefits

- **Data Caching:** Frequently accessed data can be cached at edge nodes, reducing the need to retrieve data from remote servers.
- **Reduced Bandwidth Usage:** Storing data locally minimizes the need to transmit

large volumes of data over the network.

- **Faster Access:** Local data storage ensures faster access to critical information, enhancing application performance.

1.5.3. Application Placement

Application placement services involve determining the optimal edge nodes to deploy applications based on factors like data locality, processing requirements, and user location.

Benefits:

- **Enhanced Performance:** Applications are placed on edge nodes that are closest to the data source, reducing the time taken for data transmission.
- **Efficient Resource Utilization:** Applications are deployed on edge nodes with suitable resources, ensuring efficient execution.
- **Geographical Considerations:** Application placement can be optimized based on the geographical distribution of users and data sources.

1.5.4. Real-Time Analytics

Real-time analytics services involve processing and analyzing data at the edge in real-time, enabling immediate insights and decision-making.

Benefits:

- **Instant Insights:** Real-time analytics provide immediate insights from data streams, supporting real-time decision-making.
- **Immediate Responses:** Applications can trigger immediate actions based on analyzed data, enhancing efficiency and responsiveness.
- **Reduced Data Transmission:** Only relevant insights are transmitted to remote servers, minimizing network traffic.

1.5.5. Security and Privacy

Edge computing services offer security and privacy solutions specific to edge environments, safeguarding data and applications[56].

Benefits

- **Local Data Processing:** Processing data locally reduces the exposure of sensitive information during transmission.
- **Data Encryption:** Edge services ensure data encryption, protecting it from unauthorized access.

- **Isolation and Segmentation:** Edge environments can isolate applications and data, limitingthe impact of breaches.

1.5.6. Low-Latency Communication

Edge computing services facilitate low-latency communication between edge nodes and devices, enabling real-time interactions.

Benefits:

- **Quick Response Times:** Low-latency communication ensures rapid responses for interactive applications.
- **Enhanced User Experiences:** Real-time communication supports applications like gaming,AR, VR, and telemedicine.
- **Immediate Updates:** Edge services enable instant updates and notifications for users.

1.6 Energy Management in Edge Computing

1.6.1 Definition of Energy Management

Energy management involves the systematic oversight, regulation, and preservation of energy within an organisation or system. It encompasses a methodical approach to enhancing energy efficiency by utilising technical advancements and optimisation techniques. The goal is to decrease operational expenses, improve energy reliability, and mitigate environmental consequences.

1.6.2 Relevance to Edge Computing

Energy management plays a vital role in Edge Computing, given the decentralised nature of edge devices. Edge computing greatly lowers the energy used in transmitting data to central data centres by handling data locally, either at or close to where it is generated. This move not only reduces energy usage but also lowers latency, thereby enhancing the energy efficiency of real-time processing and decision-making.

1.6.3 Benefits

Integrating energy management with edge computing offers several benefits:

Reduced Energy Costs: By processing data locally, the requirement for constant data transfer between edge devices and central servers is minimised, resulting in reduced energy consumption related to data transmission.

Lower Carbon Footprint: Edge computing helps minimise greenhouse gas emissions by optimising energy consumption, hence promoting sustainable operational practices.

Enhanced System Reliability: Edge computing decreases dependence on remote data centres by implementing localised processing, hence mitigating the effects of central system failures on energy management systems.

Improved Operational Efficiency: Processing data in real-time at the edge allows for faster reactions to fluctuations in energy demand and supply, ensuring optimal energy utilisation and minimising waste.

1.6.4 Practical Examples

Smart Grids: Smart grids leverage edge computing to rapidly adapt to fluctuations in electrical demand and supply, thereby enhancing the efficiency and stability of energy distribution.

Smart Buildings: Smart buildings utilise edge devices to optimise real-time energy consumption for various systems, including lighting, heating, and air conditioning. This results in a large reduction in energy usage without compromising comfort.

Industrial Automation: Edge computing facilitates real-time monitoring and optimisation of energy usage in industrial facilities, resulting in significant energy savings and decreased operational expenses.

Renewable Energy Management: Edge devices have the capability to efficiently control and stabilise the irregularity of renewable energy sources like wind and solar. They achieve this by promptly analysing environmental data and making necessary adjustments to system outputs.

1.6.5 Future Outlook

The convergence of AI and machine learning with edge computing is poised to significantly transform energy management as technology progresses. These technologies have the ability to forecast energy consumption patterns and make automatic adjustments to systems without the need for human involvement. This enables the development of energy management systems that are more self-governing, efficient, and environmentally friendly. Furthermore, current investigations into novel energy-efficient computing paradigms and the persistent advancement of compact, high-performance edge devices hold the potential to augment the scalability and efficacy of energy management tactics on a worldwide scale.

1.7 Conclusion

The integration of Energy Optimization Techniques (EoT) within energy-efficient frameworks represents a transformative approach in managing and reducing energy consumption across various sectors. By employing advanced algorithms and methodologies that prioritize energy savings without compromising performance, such frameworks are pivotal in addressing the global challenge of sustainable energy use. They not only enable the optimal utilization of resources but also pave the way for innovations in energy efficiency, contributing to environmental sustainability. The strategic implementation of EoT within these frameworks underscores a commitment to a greener future, demonstrating how technology can be harnessed to achieve significant energy reductions, cost savings, and a lower carbon footprint. As the world continues to move towards more sustainable energy solutions, the role of energy-efficient frameworks, enhanced by EoT, will undoubtedly become increasingly crucial in shaping a sustainable and energy-efficient global landscape.

Further, the chapters are framed out to explain the concepts of edge computing model in relation to health care. The thesis contains five chapters. Chapter two presents a comprehensive literature survey on edge computing applications in respect of the adoption of tools, the development of architecture and algorithms, model creation, and the implementation of strategies. There are numerous aspects that are strengthening edge computing implementation in complex processes and tasks and need to be reviewed for research gaps and challenge identification. All such parameters are discussed in chapter two. Chapter three will explain the problem statement and the concerned models in association with edge computing. Subsequently, a comprehensive discussion on the approach will be presented to address the identified gaps. In this chapter, the framework and architecture will be discussed to clear the strategy for improvement in the research gaps in relation to current adopted techniques. The results and discussion part will be discussed further in chapter four. The section wise outcomes and performance of algorithms and methods will be discussed to observe the improvement in the existing parameter of concern. The fifth chapter will provide a conclusion to the study on edge computing and outline the potential future directions for the research community to explore. This will involve examining the ongoing challenges related to advancements in hardware and software.

Chapter 2

Literature Survey

2.1 Introduction

The Wide Area Network (WAN) is a network that facilitates the inter-connection and linking of several Local Area Networks (LANs), providing extensive communication across diverse geographical locations, thereby enabling the long-distance transmission of data, voice, images, and videos, connecting cities, states or even countries. Advanced technologies such as Multiprotocol Label Switching (MPLS), Frame Relay, or Asynchronous Transfer Mode (ATM) networks are used to establish such connections among different LANs situated at different locations. These connections facilitate the transmission of data between different LANs, ensuring that communication is maintained, and data can be accessed in real-time regardless of the geographic distance involved. The primary function of a WAN, in fact, is to allow data to be efficiently shared and accessible by geographically dispersed branches of corporations and institutions. Its transmission speed can vary widely from 1.5Mbps (T1) to 10 Gbps (Ethernet) or can go even higher[97]. This speed depends on the transmission media, distance between the connected LANs, and the technology employed. WANs also have robust security measures such as encryption and firewalls to protect data during transmission. They also ensure that a single point of failure do not lead to complete network failure, thereby ensuring continuous network reliability and availability. As far as energy management system is concerned, WANs play an important role in establishing links between various components in a smart framework, particularly in Edge-of-Things (EoT) environments.

WAN Technologies in Edge of Things : The adoption of WAN with edge of things technology helps the originated technologies to boost the performance in saving power and improving latency[3]. The current WAN technologies such as 5G, WiFi, LoRA and LTE methods are discussed in relation to edge of things technology, which are explained as below:

1. 5G: In edge computing, 5G enables high-speed, low-latency connections, critical for real-time data processing and analysis, supporting applications like autonomous vehicles and smart cities[98].
2. 4G: provides reliable, high-speed wireless communication, supporting remote data processing and the deployment of edge computing solutions in areas lacking 5G infrastructure[99].

3. Wi-Fi 6: This technology model enables faster data transfer and supports a higher density of connected devices, facilitating efficient local data processing in edge computing environments like offices and industrial settings[100].
4. LoRa WAN: is essential for edge computing in low-power, long-range IoT applications, allowing remote sensors and devices to communicate and process data locally in areas like agriculture and environmental monitoring.
5. Bluetooth/BLE: supports short-range communication between devices, enabling local data processing and analysis in edge computing applications such as healthcare monitoring and personal fitness.
6. Zigbee: creates low-power, close-proximity networks ideal for smart home applications in edge computing, supporting communication between smart devices and aiding in home automation and energy management.
7. NB-IoT: supports low-power, wide-area connectivity, enabling edge computing solutions in remote areas and wide-area applications like smart metering and environmental monitoring.
8. Sigfox: provides connectivity for low-power, small-data devices, aiding in the deployment of edge computing solutions in applications like asset tracking and monitoring.
9. Satellite networks: ensure connectivity in remote and isolated areas, allowing the deployment of edge computing solutions in aerospace, maritime, and other remote terrestrial locations.
10. Mesh Networks: are critical for establishing resilient and robust connections in edge computing, enabling local communication and data processing in applications like disaster recovery and field operations.
11. mmWave technology provides high-capacity, low-latency communication in edge computing, supporting high-speed data transfer and applications like virtual reality in dense urban areas.
12. RFID: In edge computing, this framework supports asset tracking and inventory management by enabling quick identification, localization, and real-time data analysis in logistics and retail environments.
13. DSRC (Dedicated Short-Range Communications): is pivotal for vehicle-to-everything (V2X) communication in edge computing, enabling real-time data exchange and processing between vehicles and infrastructure, enhancing road safety and traffic management. The comparative analysis of these technologies is given below in table 2.1 to understand the need of an hour in relation to Edge of things implementation[101].

Table 2.1: Comparative analysis of WAN networks in relation to edge of things technology

Wireless Network	Range	Bandwidth	Latency	Power Consumption	Use Cases	Standardization Body
5G Networks	Long	High	Low	High	Autonomous Vehicles, AR/VR	3GPP
4G LTE	Long	Medium-High	Medium	Medium-High	Mobile Internet, Video Streaming	3GPP
Wi-Fi 6 (802.11ax)	Short-Medium	High	Low	Medium	High-Density Environments, IoT	IEEE
LoRaWAN	Long	Low	High	Low	Remote IoT Sensors, Agriculture	LoRa Alliance
Bluetooth/BLE	Short	Medium	Low	Low (BLE)	Personal Area Networks, Wearables	Bluetooth SIG
Zigbee	Short	Low	Medium	Low	Home Automation, Industrial Control	Zigbee Alliance
NB-IoT	Long	Low	Medium-High	Low	Smart Metering, Asset Tracking	3GPP
Sigfox	Long	Very Low	High	Very Low	Wide-Area IoT Applications	Sigfox
Satellite Networks	Very Long	Varies	High	High	Remote Areas, Maritime	Various
Mesh Networks	Varies	Varies	Varies	Varies	Disaster Recovery, Sensor Networks	Various
mmWave	Short	Very High	Low	High	Fixed Wireless Access, Backhaul	IEEE, 3GPP
RFID	VeryShort	Very Low	Low	Very Low	Asset Tracking, Inventory Management	ISO, IEC
DSRC	Short-Medium	Medium	Low	Medium	V2X Communications	IEEE, ASTM

In fact, cloud computing, Internet of things, and Edge of things, in recent years, have enormously amplified the significance of WANs in ensuring the seamless operations of technology because of their heavy dependence on data transfers and communications across extensive distances. The advancement of technology, however, has necessitated continuous innovations in this field to enhance speed, reliability, and security.

2.1.1 WAN integration with Smart Grids

One such innovation was the integration of Wide Area Networks (WAN) with Energy Management. Smart grids, being integral to energy management systems, have significantly evolved with this integration. WANs facilitate the optimization of smart grid operations by enabling data analytics and decision-making processes to be executed centrally, drawing upon data aggregated from diverse grid components[40]. This centralized approach enhances grid reliability by allowing for prompt detection and rectification of faults, efficient load balancing, and optimized energy distribution based on real-time demand and supply conditions. The burgeoning demands of modern power systems are being addressed, thereby enhancing operational resilience, and facilitating optimal energy utilization and distribution. Along with the facilitation of real-time monitoring and control of energy resources scattered across vast geographical areas, they also ensure an increase in the reliability and efficiency of the power system under this approach.

Not just this, they also safeguard the grid against malicious attacks and unauthorized access and protect the integrity and confidentiality of communicated data. WAN integration in smart grids has also led to further advancement in metering infrastructure. They transmit real-time consumption data from smart meters to central systems, subsequently allowing for more accurate billing, demand response management, and enabling consumers to monitor and manage their energy consumption efficiently[102].

WANs allow for seamless exchange of information between various grid components, such as substations, distributed generation systems, and energy storage systems, providing a strong communication backbone for smart grids. This integrated communication capability is fundamental in ensuring the coherence and stability of smart grid operations, especially in managing distributed energy resources and mitigating potential disruptions. The integration of WANs in smart grids is also crucial in managing renewable sources of energy like solar and wind power. The real-time data communication facilitated by WANs enables efficient balancing of supply and demand, ensuring optimal utilization of renewable energy and enhancing the sustainability of the power system. Thus, this WAN integration in smart grids also evolved energy management practices to an extent, advancing the capabilities and resilience of modern power systems.

Zhiqiang et al. (2023) developed a very sensible method for accessing the terminal information to a remote location where the monitoring system was installed to measure the health-related parameters. This model was very well versed in processing complex data forms in a very limited number of job scheduling task lines. For transmission of data, an E- model was proposed for lower-end data communication between the modules. This developed methodology promotes remote location accessibility for the establishment of low-power communication gates between transceiver modules[103].

2.1.2 WAN integration with Edge of Things

Apart from smart grids, WAN can also be combined with edge computing devices. This amalgamation ensures that the data must be processed seamlessly across vast geographical locations, subsequently leading to more responsive, efficient, and reliable energy management solutions. On one hand, where Edge-of-Things (EoT) allows for decentralized data processing, on the other, WANs ensure seamless connectivity among various edge devices and the central system, enabling prompt decision-making and real-time data analysis. The integration of EoT and WAN fosters energy efficiency by enabling smart energy management solutions at the edge of the network[104]. These solutions can monitor and control energy consumption in real-time, optimizing the usage of energy resources and reducing operational costs. This integration also offers enhanced scalability and flexibility in energy management along with increased reliability and responsiveness in the system. WANs ensure uninterrupted connectivity between edge devices and central systems, while EoT facilitates rapid data processing and decision-making at the network edge, reducing response times in energy management operations. Even though the integration of EoT and WAN is revolutionizing energy management system and offers plethora of benefits, but at the same time it also necessitates rigorous cybersecurity and data privacy measures because of increased vulnerability due to its extensive connectivity and decentralized nature of data processing[10].

2.1.3 WAN Technologies in Renewable Energy

The deployment of WAN in renewable energy sources have also gained popularity in recent years. WAN technologies enhance the scalability and accessibility of renewable energy systems, allowing for improved management and distribution of energy resources. The integration of Wide Area Networks (WAN) technologies in renewable energy systems is instrumental in ensuring the optimal, resilient, and efficient management of diverse energy sources[105]. WAN technologies are central in addressing the complexities and demands inherent to renewable energy systems, fostering advanced energy solutions, and contributing to the sustainability of energy ecosystems. WAN technologies enable the real-time monitoring and control of renewable energy sources, such as solar and wind power, dispersed over vast geographical areas. This real-time capacity not just optimizes the overall energy production but also mitigates the energy losses and ensures a reliable and stable renewable energy system.

WAN technologies empower renewable energy systems with advanced data analytics capabilities[106]. They allow for the centralized processing of diverse and extensive datasets, enabling energy providers to make informed and optimized decisions regarding energy production, distribution, and consumption. It also provides accurate and real-time energy consumption data, which is crucial in energy forecasting, precise billing, and demand response management. This integration of WAN technologies helps in connecting renewable energy systems to the grid, which enables smooth, efficient, and uninterrupted communications between them, which in turn allows for an efficient energy balancing and distribution system.

They facilitate the interconnection of diverse energy systems, allowing for the adaptive and scalable expansion of renewable energy infrastructures, thereby providing for an essential backbone to the system. Taking the extensive and interconnected nature of WAN-integrated renewable energy systems into consideration, it becomes highly important to provide robust cyber security measures in order to protect the integrity, availability and confidentiality of data and systems[78]. This WAN integration is paramount in addressing the challenges posed by the intermittent and diversified nature of renewable energy sources and is crucial in advancing the capabilities, resilience, and sustainability of modern renewable energy systems.

2.1.4 WAN and Smart Healthcare

Khanh et al. (2023) observed that the modern era is totally lying on edge computing-based modules for caring the health of human subjects. The E- health care strategy is one of the leading research areas now a days for development of new applications with local processing for decision taking ability of machine itself. The team identified certain issues in relation to cloud computing such as slow rate of processing, decision taking capability or high level of responsiveness time for execution of target nodes. In this regard they proposed a queue-based network model for bridging the gap between said issues to improve the response time with high level of accuracy and power management in the edge devices[107].

Chakraborty et al. (2023) did research on cyber security and the protection of data on edge devices. It was observed that the cloud has an open source for data manipulation and that attackers can malpractice with the transmitted data in certain ways. So, a Multi-Source Transfer Learning system that was operated centrally to protect the transmission of data on edge devices was developed along with an AI-based framework for the protection of data in the middle of paths to prevent attackers and hackers from doing anything with the data on edge devices and applications. When the developed model was tested with the EMNIST, X-IIoTID, and Federated TON_IoT data sets, the framework latency improved[108].

Singh et al. (2023) also proposed a framework to protect the data between edge devices. It was found that the Edge of Things is a middle layer between the end user and the cloud, and the EoT is the only tool to reduce the processing time between the nodes and modules for improvement in the overall system. A very secure framework was developed with attribute-based encryption and cluster-based processing of data to improve the quality of data on edge devices. During testing, more than 90% accuracy was achieved on the edge systems[109].

Upadhyaya et al. (2023) did a healthy research survey and presented very relevant issues that persist in edge of things communication. In health care systems, architecture development is one of the key research areas to focus on for the establishment of efficient and energy-saving applications. Real-time data collection methods and analysis tools for power-saving in edge device communication were also stressed upon in the study. The importance of edge AI tools was also highlighted for solving many problems in this regard[110].

2.1.5 IoT and WAN for Energy Management

WANs are also integrated with IoT devices to facilitate an uninterrupted exchange of information, ensuring interoperability among the connected devices, contributing to smarter and more efficient energy management[97]. The integration of Internet of Things (IoT) with Wide Area Networks (WAN) is at the forefront of transforming energy management solutions, providing a new pathway to more efficient, sustainable, and intelligent energy systems. This union enables the harmonization and streamlining of vast and diverse data from IoT devices, leveraging WAN's extensive connectivity to offer advanced energy management capabilities. Combining IoT and WAN provides unparalleled monitoring and control capabilities to energy systems, which allows real-time data processing, acquisition, and response. This integration contributes to the overall sustainability of the system by providing enhanced consumption, optimizing energy efficiency, and reducing energy wastage[111].

This also balances the energy supply and demand by preventing energy shortages and excesses, thereby optimizing overall energy distribution and pricing system. This also enables energy demand forecasting and adaptive energy supply modulation. The fusion of IoT and WAN technologies contributes significantly to optimizing grid operations and enhancing grid reliability. It seamlessly communicates and coordinates between various energy sources, storage systems, and loads, ensuring the stability and reliability of the energy grid. IoT devices, when connected through WAN, can also offer high maintenance for energy systems by constantly monitoring the equipment and predicting potential failures even before they occur. Not just does it reduce the downtime, but also lowers the maintenance cost thereby extending the overall lifespan of the equipment. It can further aid in identifying energy inefficiencies and implementing corrective measures, leading to energy conservation and cost reduction[112].

This combination of IoT and WAN allows for the detailed analysis of energy consumption patterns and facilitates the development of strategies to optimize energy usage. Thus, it can be said that the combination of IoT and WAN in energy management is revolutionizing the way energy is monitored, controlled, and optimized. This integration promises enhanced energy efficiency, improved grid reliability, advanced demand response management, and optimized energy consumption, while also emphasizing the importance of implementing robust security measures to protect data and privacy in interconnected energy systems[113].

2.1.6 Security concerns and WAN

There are numerous benefits being offered by the integration of WAN in energy management systems, however, security remains one of the most significant concerns. Due to their extensive reach and connectivity, WAN technologies are prone to network intrusions and attacks such as DDoS attacks, which can incapacitate energy management systems, disrupt energy supply, and compromise the stability of energy grids[114]. With the vast amounts of sensitive data transmitted over WAN, energy management systems are susceptible to data breaches and information leaks. Also, Unauthorized access and extraction of energy data can have severe repercussions, compromising user privacy and system reliability.

Robust security measures are important to protect sensitive data and maintain the integrity of energy management systems. Moreover, the deployment of WAN technologies in energy management systems exposes them to malware and ransomware threats. This malicious software can corrupt system data, damage critical infrastructure, and extort organizations for financial gains. In relation to EoT based safeguard methods, the extensive and interconnected energy management systems, a few comprehensive security measures can be adopted. One of them is ensuring robust authentication and authorization to protect unauthorized access and malicious software. WAN integration also necessitates to encrypt data and secure communication channels in order to protect confidentiality and integrity of information transmitted. Regulatory standards here plays a vital role while integrating WAN technologies in energy management, the non-compliance of which can lead to legal repercussions, financial penalties, and can compromise the trust and reliability placed in energy management systems[78], [115].

Gai et al. (2021) observed that cyber-attacks can be of two types, virtual and physical. In the era of IoT, data is passing on from various remote devices at a time, and in this regard, physical attacks on the data cannot be tracked for identification of source and impact intensity. So, they proposed a framework with true random number generators for the prevention of these attacks. This cluster-based method was very efficient in protecting the data from technical glitches[116].

Saheed et al. (2022) observed that the classification of data is one of the prime tools to make efficient and fast-edge applications. In this regard, they proposed and improved classification methods, such as component analysis and gradient-based boosting algorithms, to improve the quality of the system in power-saving mode. As per verification of results, more than 92% efficiency was observed in the system[117].

Edge computing (EC) is a computing methodology that is distributed in nature that brings data storage and computation closer to the place where it is to be used to accelerate response time and the bandwidth savings. The Internet of Things (IoT) refers to the collection of all those devices that could connect to the internet to collect and share data[118]. It is a serious problem to safeguard the IoT environment using a traditional intrusion detection system (IDS) due to the diverse types and huge number of IoT devices. The architectural change in the Edge of Things (EoT) causes privacy and security problems to migrate to dissimilar layers of the edge architecture[119], [120]. Therefore, detecting intrusion attacks in a distributed environment as such is problematic. In this situation, an IDS is required. This research group proposed improved IDS models for the classification of attacks on IoT and EoT. To protect EoT and IoT appliances and devices, an improved IDS-IoT was developed by implementing nine different machine learning models. The normalization technique was performed using the minimum-maximum (min-max) method[121]. Subsequently, dimensionality reduction was performed with Principal Component Analysis (PCA). The light gradient boosting machine, decision tree, gradient boosting machine, k-nearest neighbor, and extreme gradient boosting algorithms were used for classification.

2.1.7 Network Optimization in Energy Management

In contemporary energy management systems, network optimization thus plays a pivotal role in enhancing the efficiency, reliability, and sustainability of energy distribution and consumption. Optimal network configurations significantly reduce energy consumption and operational costs. Through dynamic routing and bandwidth allocation, WANs can be customized to meet the specific needs of energy management systems. This helps in achieving an effective load balance by evenly distributing energy loads across different pathways. By optimizing energy distribution and reducing energy losses, network optimization contributes to significant cost savings for energy providers and consumers[21]. It allows for dynamic pricing models, encouraging energy conservation and enabling consumers to manage their energy consumption cost-effectively.

Network optimization also ensures that the available energy resources should be optimally used in such a way that all the varying demands across different locations and times are efficiently met. They can analyze energy consumption patterns and forecast future demand accurately which allows energy providers to modulate energy supply proactively, ensuring that energy production is aligned with consumption needs. Network optimization seamlessly integrates renewable energy sources like solar and wind energy with the energy grid to reduce heavy dependency on non-renewable sources and to promote environmental sustainability[122]. Along with enhancing the stability and reliability of energy grids, it also reduces the risk of outages and disruptions. Furthermore, the optimization of network protocols enables real-time monitoring of energy systems, which helps in detecting anomalies and managing efficient energy flows, ultimately contributing to the advancement of intelligent and sustainable energy management solutions.

2.1.8 Challenges in WAN-Integrated Energy Management

Despite its numerous benefits, the integration of WANs in energy management systems is fraught with a lot of challenges. These challenges revolve around the complexities of network architecture, security concerns, data management, latency, and resource allocation and can impact the overall performance and reliability of energy management systems. The vast geographical spread of networks can highly impact the real-time-monitoring and control of energy systems, thereby causing latency and network delays. These delays, in turn, can compromise the reliability and responsiveness of energy management applications. The extensive connectivity of WAN also elevates the risk of unauthorized access, data breaches, and cyber-attacks, necessitating advanced security mechanisms and protocols to safeguard sensitive information and maintain user trust thereon[123]. Handling enormous volumes of data effectively is another challenge. The need to process, analyze, and store vast amounts of energy data generated from diverse sources demands robust and scalable data management solutions, absence of which can lead to information overload and compromise decision-making processes. Any network downtime or unavailability can disrupt energy distribution and monitoring services, affecting both energy providers and consumers adversely.

Thus, it becomes highly important to ensure consistent network availability for uninterrupted energy management operations. WAN integration also offers inherent complexity in integrating diverse and incompatible systems. This needs sophisticated strategies and solutions to ensure seamless operation and optimization of energy management system. Allocating efficient network resources and optimizing energy distribution are another set of challenges that may lead to energy wastage, reduced system performance, and increased operational costs[124]. Thus, it becomes highly imperative to have innovative solutions and continuous advancements in technology to overcome these challenges in WAN-integrated energy management and addressing these challenges is crucial for realizing the full potential of WAN-integrated energy management systems and achieving sustainable energy solutions[125].

2.2 Energy Management

Energy management systems (EMS) in the Edge-of-Things (EoT) paradigm is another approach which aims to optimize the efficiency, reliability, and sustainability of energy resources by leveraging the edge computing model. The EoT allows for real-time data processing at the network edge, closer to the location where it is needed, reducing the latency, and allowing faster responses. Several studies have been conducted focusing on the implementation of EoT for optimizing energy resources. Integrating EoT in Energy Management Systems (EMS) enables enhanced monitoring and control of energy resources, improving efficiency and reducing energy consumption[112]. EoT also enables real-time processing of data from various energy resources, allowing for instantaneous analysis and decision-making. This is crucial for not only optimizing energy consumption but also responding swiftly to changes in energy demand or supply, to meet the fluctuating needs of energy. Energy Management Systems helps in redistributing energy loads and adjusting energy supply based on real-time demand, thereby achieving optimal load balancing and effective demand response. This not only stabilizes the energy grid but also enhances energy efficiency and reduces overall operational costs. EoT helps in reducing reliance on non-renewable energy and promoting sustainability by facilitating seamless integration of renewable energy sources into the energy grid. The implementation of Edge of Things (EoT) enables Energy Management Systems (EMS) to effectively improve energy efficiency by employing intelligent routing and control mechanisms, hence optimizing energy distribution, and minimizing energy losses[126]. This phenomenon leads to substantial reductions in energy use and additionally contributes to the preservation of the environment. In fact, the implementation of Edge of Things (EoT) empowers Energy Management Systems (EMS) to anticipate prospective equipment malfunctions and proactively arrange maintenance activities thereon. This not only reduces the amount of time that energy equipment is not in operation, but also prolongs its overall life span and guarantees a reliable and uninterrupted energy supply. Thus, by addressing various aspects of EoT integration with EMS, more resilient, sustainable, and efficient energy management solutions can be developed[106].

2.2.1 Real-Time Data Processing and Analytics

The incorporation of Edge of Things (EoT) in Energy Management Systems (EMS) has significantly enhanced the capabilities of real-time data processing and analytics. This advancement has pivotal implications for real-time management, allocation, and conservation of energy resources. EoT also provides advanced analytics capabilities to EMS, allowing for the extraction of meaningful insights from complex energy data sets. These insights enable energy providers to make informed decisions, optimize energy usage, and enhance service delivery[127]. Through the utilization of real-time data analytics provided by EoT, EMS can optimize energy consumption patterns, identify inefficiencies, and implement corrective measures instantaneously. This result is helpful in enhanced energy efficiency and substantial cost savings. Employing EoT enables the proactive identification of possible faults and the enhancement of maintenance schedules. This helps in the reduction of operational downtime and the maintenance of a continuous and uninterrupted energy supply. Real-time monitoring and control of energy systems are established by EoT, thereby allowing for immediate response to changes in energy demand or supply. This helps in maintaining grid stability, optimizing energy distribution, and preventing energy wastage. The EoT platform provides EMS (Energy Management Systems) with scalable and adaptable solutions that can effectively adjust to different energy demands and operational conditions. The capacity to do data processing at the edge facilitates the incorporation of various energy sources and facilitates the scalability of energy networks[112]. The integration of EoT in Energy Management Systems for real-time data processing and analytics has resulted in substantial advancements in energy management.

2.2.2 Renewable Energy Sources and EoT

In order to effectively and sustainably address the changing requirements of energy consumption, the integration of renewable energy sources into Energy Management Systems (EMS) is becoming increasingly crucial, with Edge of Things (EoT) playing a significant part in this endeavor. The integration of Edge of Things (EoT) with renewable energy sources facilitates the development of intelligent and autonomous energy management solutions. It adjusts the storage of energy generated from renewable sources based on consumption patterns and demand forecasts[128]. The utilization of advanced analytics capabilities by the Edge of Things (EoT) facilitates the precise prediction of energy generation derived from renewable sources. Accurate forecasting plays a pivotal role in facilitating efficient energy management by enabling the optimization of energy supply to align with demand and mitigate energy inefficiencies. It seamlessly integrates with diverse renewable resources such as solar, wind, and hydropower, into the energy grid, allowing for optimized and balanced energy supply. This ensures the availability of stored energy during peak demand periods and mitigates the impact of renewable energy intermittency. EoT also facilitates the management and optimization of Distributed Energy Resources (DERs), enabling decentralized energy production and consumption, which contributes to the reduction of energy transmission losses and enhances grid reliability[129]. Thus, by leveraging EoT in the management of renewable energy sources, EMS can attain heightened sustainability by diminishing the release of greenhouse gas emissions, enhancing energy efficiency, and advocating for the overall preservation of energy[23].

Popli et al. (2021) observed that the energy transmission models consume a huge amount of power in EoT-based modes for processing and execution of target nodes in relation to decision-making situations. In this regard, emphasis was placed on 5G-based communication models for power saving. They proposed a 5G-based framework for the establishment of EoT applications. They found that their model was faster than primitive models but saved only 20% more energy than the existing models[130].

Singh et al. (2023) observed that AI is one of the best tools to manage the database on the EoT device network to improve the performance of the model. Machine learning and deep learning-based algorithms are two of the main tools in data management, and these tools also help sort the data for processing and stacking. The AI-based tools help sort the data based on various features, and as per the rank of the features, the training of the data has been done to map the concerned output for processing and the decision of the machine itself[131].

Cao et al. (2023) observed that the multi-objective grey wolf-based optimization algorithm in relation to Unmanned Aerial Vehicles (UAVs) is one of the powerful tools to improve the quality of EoT-based devices in terrestrial IoT network applications. Their optimization method was very efficient in improving the overall performance of the system[132].

2.2.3 Machine Learning in EoT-based EMS

The incorporation of Machine Learning (ML) into the Edge of Things (EoT) has further advanced a transformative approach to improve energy efficiency, decision-making processes, and resource optimization. The integration of machine learning techniques in EoT-based EMS showcases how predictive analytics and intelligent algorithms can optimize energy consumption and reduce operational costs. It enables predictive maintenance by analyzing real-time data to anticipate equipment failures and schedule timely maintenance, thus reducing downtime and operational costs[133]. Not just this, when deployed at the edge, ML algorithms empower EMS with the ability to predict accurate energy consumptions for optimal energy distribution. The integration of machine learning (ML) with the Edge of Things (EoT) enables an efficient balance of supply and demand. It facilitates the optimization of demand response methods through the analysis of consumption patterns and the real-time adjustment of energy supply. It also facilitates the detection of anomalies in energy consumption patterns, enabling the identification of energy inefficiencies and unauthorized energy usage, thus improving energy security and efficiency[134]. Furthermore, EoT- based EMS can minimize reliance on non-renewable sources by applying ML to analyze weather conditions and other relevant factors, thereby maximizing the energy harvest. Thus, this renewable energy optimization, ML in EoT-based EMS contributes significantly to the sustainable and efficient use of energy resources.

2.2.4 Challenges and opportunities in EoT-based EMS

While EoT brings forth significant advancements and numerous opportunities, the aspects of scalability and security in EoT-based EMS continue to pose substantial challenges. EoT-based Energy Management Systems (EMS) are critically essential for intelligent energy management and optimal resource utilization, contributing to environmental sustainability (Table 2.2).

Table 2.2: Issues associated with EoT-based EMS

S No.	Current Issues	Opportunities	Challenges
1.	Scalability Issues	Scalability issues have ignited the development of novel scalable architectures and algorithms, which can manage large volumes of data efficiently, thereby ensuring the seamless expansion of EoT-based EMS.	With the rapid expansion and deployment of IoT devices, ensuring the scalability of EoT-based EMS is daunting. Managing extensive data generated from myriad sources requires robust and scalable solutions.

2.	Security Concerns	The prevalent security challenges necessitate the advent of innovative security protocols and encryption methods, ensuring the confidentiality, integrity, and availability of the information in EoT-based EMS.	The integration of multiple devices and the vast amount of data generated expose EoT-based EMS to various security threats, including data breaches and unauthorized access.
3.	Data Integrity and Privacy	The challenges related to data integrity and privacy are driving the research and development of advanced data integrity verification methods and privacy-preserving techniques in EoT-based EMS.	Maintaining the integrity of data and ensuring the privacy of user information in EoT-based EMS are significant hurdles due to the diverse nature of data sources.
4.	Resource Constraint Issues	This limitation has paved the way for optimization techniques and lightweight algorithms to maximize the resource utilization efficiency of EoT devices, ensuring optimal performance.	EoT devices usually operate under constrained resources, which restrict their processing capabilities and affect the overall performance of the EoT-based EMS.

While scalability and security issues in EoT-based EMS pose significant challenges, they concurrently act as catalysts for innovations and advancements in scalable solutions, security protocols, data integrity verification methods, and resource optimization techniques, enhancing the overall efficacy and reliability of EoT-based EMS[112], [135].

2.2.5 Emerging Trends and future Directions

Furthermore, emerging trends such as the incorporation of Artificial Intelligence (AI) and the Internet of Things (IoT) in EoT-based EMS offer promising prospects. EoT-based Energy Management Systems (EMS) are at the forefront of technological innovation, playing a pivotal role in fostering energy efficiency and sustainability. The integration of cutting-edge technologies and methodologies is paving the way for a myriad of emerging trends and future directions, each posing its unique set of challenges and opportunities as given below (Table 2.3).

Table 2.3: Emerging Trends Opportunities and Challenges

S No.	Emerging Trends	Opportunities	Challenges
1.	Integration of 5G Technologies	5G technologies facilitate enhancedconnectivity and low-latency communication, fostering the development of advanced, responsive, and real-time energy management solutions.	However, the integration of 5G brings forth challenges related to network security, interoperability, and standardization, necessitating the development of robust security protocols and interoperable solutions.
2.	Blockchain for Secure Transactions	The decentralized nature of blockchain enhances security and trust, enabling secure energy trading and transparent energy transactions.	However, scalability and energy consumption of blockchain networks pose substantial challenges, driving the need for optimized and energy-efficient blockchain solutions.
3.	AI and Machine Learning for Advanced Analytics	The incorporation of AI and ML offers unparalleled opportunities for predictive maintenance, demand forecasting, and optimization, enhancing the efficiency and reliability of energy management systems.	The implementation of AI and ML necessitates addressing challenges related to data privacy, model transparency, and ethical considerations, emphasizing the importance of ethical AI development and deployment.

The convergence of 5G, blockchain, and AI/ML in EoT-based EMS signifies a transformative phase, driving the development of innovative and advanced energy management solutions. While each trend holds immense promise, addressing the inherent challenges related to security, scalability, energy consumption, data privacy, and ethics is paramount to realizing the full potential of these emerging technologies in energy management[136], [137], [138].

Gadekallu et al. (2021) conducted a research survey to integrate IoT and EoT devices using blockchain methods. They surveyed the literature to explore the new dimensions for convergence in application-based modules for enabling the end user to reach the maximum area-specific modules for sharing the data and processing the node-based data for quick and accurate decisions[139].

Akhunzada et al. (2023) presented various leading-edge computing-based frameworks to improve the efficiency of networks with limited use of power. In this regard, they tested and compared various wireless transmission modules for energy efficiency and network security. They also worked on the utilization of power for AI-based optimization frameworks to manage the data and reduce the real-time delay in processing. In hardcore industrial applications and military-based tasks, their analysis helped to set the benchmark for the adoption of power-saving modules[140].

2.3 Job Scheduling and Resource Allocation

Another important aspect in the realm of computing and information and technology is job scheduling and resource allocation, significantly impacting the performance, efficiency, and functioning of systems. They play an important role in optimizing the utilization of resources, ensuring timely execution of tasks, and maintaining the balance in workloads. Efficient allocation and scheduling are crucial to manage the finite resources in EoT environments effectively.

- **Job scheduling** refers to the practice of assigning processes and tasks to processors or computing resources in an optimal manner, with the overall aim of improving performance, minimizing delay, and optimizing resource usage. The system integrates a variety of algorithms and methods, including First Come First Serve (FCFS), Shortest Job First (SJF), and Round Robin (RR), each with distinct advantages and applications. The evolving nature of job scheduling is accentuated by advancements in AI, allowing the development of intelligent and adaptive scheduling algorithms, capable of learning and evolving with changing system dynamics and workloads[141], [142].

- **Resource allocation** on the other hand, is pivotal for enhancing the efficacy and efficiency of computing systems, focusing on the optimal distribution of available resources such as CPU, memory, and bandwidth among competing tasks and processes. It’s a multifaceted process involving the determination of resources to be allocated, decision-making on resource distribution, and the implementation of allocation policies. Resource allocation in cloud and edge computing environments is particularly challenging, demanding meticulous strategies to balance the trade-off between resource availability and task requirements[143].

The amalgamation of job scheduling and resource allocation is important in order to fulfill the escalating requirements for enhanced performance and efficiency in modern computing settings. Through, the process of synergizing these many components, systems may effectively guarantee the optimal allocation of resources. Additionally, systems can assure efficient utilization of these resources by aligning the scheduling of jobs with the available resources. Apart from opening up new opportunities, the continuous evolution of computing paradigms, including cloud, edge, and fog computing, is posing new challenges in job scheduling and resource allocation. The emergence of technologies like IoT and Edge-of-Things (EoT) necessitates the development of advanced strategies to address the complexities and dynamics of these environments[144]. Here, advanced optimization and machine learning techniques are playing a pivotal role in addressing these challenges by enabling adaptive and intelligent resource allocation and job scheduling strategies. Thus, Job scheduling and resource allocation serve as the backbone of computing systems, playing a crucial role in achieving optimal performance and efficient utilization of resources. The advancements in technology and the integration of AI and machine learning are shaping the future of these domains, promising enhanced adaptability, efficiency, and intelligence in managing resources and scheduling jobs in diverse computing environments[145].

2.3.1 Integration in EoT-based EMS

The integration of Edge-of-Things (EoT) in Energy Management Systems (EMS) has witnessed the incorporation of advanced optimization strategies to address the multifaceted challenges of energy conservation, distribution, and consumption in real-time. Optimization strategies in EoT-based EMS revolve around the efficient management and allocation of energy resources, exploiting the capabilities of edge computing to process data closer to the source, thereby reducing latency and improving response times. The implementation of real-time optimization algorithms is crucial in order to fully use the capabilities of Edge of Things (EoT)-based Energy Management Systems (EMS). These facilitate real-time data processing and decision-making, enabling dynamic adjustments to energy consumption and distribution in response to fluctuating demand and supply, thereby ensuring efficient application of energy resources. Another set of optimization strategy includes machine learning and artificial intelligence (AI), which play an important role in allowing the EMS to learn from historical data, predict future trends, and make intelligent decisions. These technologies empower EoT- based EMS to adapt to changing environments, optimize energy consumption patterns, and enhance overall system efficiency[78].

Demand response optimization is another strategy for effective demand response allowing for the modulation of energy consumption based on real- time supply and demand conditions. This not only aids in balancing the load but also in mitigating the risks of energy shortages and ensuring uninterrupted power supply. These optimization strategies also focus on the seamless integration of renewable energy sources, like solar and wind, into the energy mix. The advanced optimization strategies in EoT-based EMS are shaping the future of energy management by ensuring real-time, intelligent, and efficient energy resource allocation and utilization. The integration of AI and machine learning with real-time processing capabilities of EoT offers unprecedented opportunities for optimizing energy consumption, integrating renewable energy sources, and enhancing the overall sustainability of energy ecosystems[115].

2.3.2 Distributed Computing Environments

The amalgamation of distributed computing environments and EoT-based EMS is another approach which provides several advantages including enhanced scalability, reliability, and efficiency in energy management. Adaptive scheduling algorithms can efficiently allocate resources in real-time, catering to the dynamic needs of EoT-based EMS. Distributed computing environments are fundamental for the integration in Edge-of-Things (EoT)-based Energy Management Systems (EMS) as they empower these systems to process and analyze data in real-time at different locations, enabling prompt and intelligent decision-making. Distributed data processing reduces the time taken to transmit data to a centralized location for processing as it processes data at various points in the network[146]. This localized processing enables faster response times and more efficient energy utilization. Distributed computing environments also provide EoT-based EMS with the scalability and flexibility required to adapt to varying workloads and energy demands. They facilitate the addition of new resources and devices to the system with minimal disruption, allowing for seamless expansion and contraction of computing resources in response to changing energy requirements. Moreover, the inherent redundancy within distributed computing environments serves to improve the dependability and fault tolerance of EoT-based EMS. In the event of a node failure, the system has the capability to maintain regular operation by redirecting tasks to other operational nodes, hence guaranteeing continued provision of energy management services.

Apart from this, effective load balancing also serves as a key attribute to distributed computing in EoT-based EMS. It ensures an equitable distribution of tasks and workloads across multiple nodes, preventing over burdening of individual nodes and optimizing overall system performance and energy efficiency[112]. The incorporation of distributed computing environments in EoT-based EMS plays a crucial role in attaining effective, scalable, and dependable energy management. The decentralized character of these environments facilitates the processing of data in specific locations, which is essential for the timely management of energy and decision-making. Additionally, their ability to scale and tolerate faults allows for smooth adjustment to fluctuating energy requirements and uninterrupted operation in the event of system failures[135].

Zhong et al. (2020) observed that the heuristic approach is one of the best approaches to managing the database to optimize the use of data centers for data storage and scheduled maintenance. It was observed that a huge amount of money is spent on making data centers and installing hardware models for data storage. They found that the Containerized Task Co-Location (CTCL) scheduler is one of the best optimizers to manage the data[147].

Khatua et al. (2023) found that the wireless modules are facing internet connectivity issues while synchronizing the modules for data sharing and passing on the decisions to the end user. It has been observed that 65% of end users were able to make the final decision due to poor connectivity with remote operators. As such, the files were sent back to the cloud server, creating a heavy rush to transmit packets. With the adoption of novel dew-caching architecture under the cloud using the Internet of vehicular things (IoVs), the issue is resolved to some extent and saves resources[148].

2.3.3 Machine Learning Approaches

Recent studies have shown the convergence of Machine Learning (ML) and Edge-of-Things (EoT) in Energy Management Systems (EMS) is another approach which represents a progressive step towards the creation of intelligent, adaptable, and predictive energy management solutions. ML approaches within EoT-based EMS focus on leveraging the abundance of data generated by IoT devices to make informed decisions related to energy consumption, distribution, and conservation. The utilization of ML in EoT-based EMS enables the development of predictive models that forecast energy consumption patterns and demand based on historical data[134]. This predictive analytics capability allows for proactive energy management strategies, reducing energy wastage and optimizing resource allocation. Machine learning (ML) methodologies play a crucial role in Edge-of-Things (EoT)-based Energy Management Systems (EMS) by facilitating the detection of anomalies and enabling effective system monitoring.

Real-time monitoring systems enable the prompt detection of irregularities and potential problems in the energy infrastructure, so enabling urgent corrective measures to be taken. This ensures the dependability and stability of energy distribution networks. ML techniques also optimize the integration and utilization of renewable energy sources by predicting their availability and output, based on environmental conditions and historical data. This enables the effective balancing of energy supply from renewable and non-renewable sources, promoting sustainability and reducing dependency on fossil fuels[146]. Not just this, ML approaches also facilitate advanced demand response management in EoT-based EMS, allowing for real-time adjustments to energy consumption, external factors, and pricing models, ensuring efficient energy utilization and cost- effectiveness. Thus, Machine Learning approaches in EoT-based EMS symbolize a new era of intelligent energy management, thereby contributing to the sustainability and efficiency of energy ecosystems in the context of distributed and edge computing environments.

Le Nguyen et al. (2023) adopted a machine learning-based Shapley additive feature selection method to improve the efficiency of EoT applications. They also segregated the data based on rank of feature using ANN-based methods and then tested the energy efficiency model for predicting the shear strength of RC deep beams with the edge processing method. They used seven AI features and classification methods such as linear regression, artificial neural networks (ANN), support vector machines, decision trees, ensembles of trees, extreme gradient boosting, and gaussian process regression, and they found that the gaussian process regression had the best response among the other methods[149].

Xu et al. (2023) observed that the Synthetic Minority Oversampling Technique (SMOTE) algorithm and mutual information sharing are among the best optimization methods to calculate the hyperparameters for EoT devices. This classification method was observed to be helpful in solving multi-class classification problems with an accuracy of more than 90%, which results in outperforming the existing algorithms by a decent margin[150].

2.3.4 Challenges and solutions in Job Scheduling and Resource Allocation

2.3.4.1 Complexity and Scalability Issues

Despite the advancements in implementing effective scheduling and allocation strategies, significant challenges persist. One significant challenge is managing the complexity and ensuring scalability to accommodate the growing demand and the heterogeneous nature of resources available at the edge of the network. The author proposed solutions to address these challenges, focusing on the development of scalable algorithms and frameworks to manage these growing complexities in EoT-based EMS[112]. The presence of diverse task needs, resource limits, and the dynamic nature of the environment generally gives rise to complexity in work scheduling and resource allocation. The effective utilization of resources and work scheduling in EoT necessitates the implementation of advanced algorithms that can effectively handle the complexities arising from various conflicting objectives and limitations. Modern solutions incorporate advanced optimization techniques to navigate through the complexities, ensuring that the right resources are allocated to the right tasks at the right time.

Ensuring Scalability is another challenge and a crucial aspect of addressing the expansion of EoT-based EMS. It is imperative to design systems capable of adapting to increasing loads, diverse task types, and the continuous addition of new resources. And for that, Strategies such as decentralized scheduling and hierarchical resource allocation have been proposed to enhance scalability, enabling the system to manage larger sets of heterogeneous resources efficiently[151]. To counter complexity and scalability issues, several solutions have been proposed, which includes adaptive scheduling algorithms that possess the capability to adaptively modify their behavior in accordance with the evolving environment and the availability of resources. Furthermore, the integration of machine learning and artificial intelligence has been implemented to properly forecast resource requirements and make educated decisions regarding scheduling, hence optimizing the utilization of resources and enhancing the overall performance of the system. Hence, Complexity and scalability are paramount concerns in job scheduling and resource allocation in EoT-based EMS. The solutions to these concerns lie in the development of advanced, adaptive, and intelligent algorithms and strategies that can efficiently navigate through the complexities and scale according to the growing and diverse demands of the EoT environment[135].

2.3.4.2 Dynamic and Heterogeneous Environment

The amalgamation of Edge-of-Things (EoT) in Energy Management Systems (EMS) creates environments that are dynamically evolving and extremely heterogeneous, imposing substantial challenges to job scheduling and resource allocation. As the intricacy and diversity of tasks and resources intensify, it becomes imperative to develop solutions that are adaptive, robust, and efficient. Dynamic environments in EoT-based EMS are characterized by frequent changes in resource availability, task requirements, and workload conditions. Traditional scheduling and allocation models often struggle to adapt to these variations, leading to suboptimal resource utilization and decreased system performance. Adding to this, Heterogeneity poses another significant challenge, stemming from the diversity in resource types, capabilities, and constraints[152].

Efficiently managing heterogeneous resources necessitates intelligent and flexible algorithms capable of considering the unique attributes and requirements of each resource and task. Here, to address the challenges posed by dynamic and heterogeneous environments, researchers have put forth adaptive scheduling and allocation solutions, which utilize real-time data and advanced analytics to dynamically adapt resource allocation and work scheduling in accordance with environmental fluctuations, thereby guaranteeing optimal system performance and resource usage. Apart from these, Machine Learning (ML) and Artificial Intelligence (AI) techniques are also being increasingly incorporated to address the complexities arising from dynamic and heterogeneous EoT environments[29]. These techniques enable predictive analytics, intelligent decision-making, and automated adaptation to the changing conditions and requirements, thereby enhancing the efficiency and robustness of the EMS.

2.3.5 Emerging Trends and Technologies

2.3.5.1 AI-Driven Scheduling and Allocation

Artificial Intelligence (AI) is revolutionizing the domain of scheduling and allocation within the Edge-of-Things (EoT) in Energy Management Systems (EMS) to overcome the substantial challenges imposed because of complexity of the dynamic and heterogeneous EoT environments. It brings about enhanced automation, efficiency, and intelligence, which are pivotal in optimizing the intricate and dynamically changing environments inherent to

EoT- based systems. AI-driven scheduling and allocation leverage sophisticated algorithms and models, such as Machine Learning (ML), enables systems to learn and adapt continuously to changing conditions, optimizing resource utilization and reducing operational costs. One of the prime benefits of incorporating AI is the provision of predictive analytics, which can forecast future demands, resource availability, and potential system failures[153]. Predictive analytics aid in proactive decision-making, allowing for the anticipation of changes and adjustment of schedules and allocations beforehand to avoid disruptions and maintain optimal performance. Not only this, AI-driven solutions can rapidly and accurately respond to dynamic changes in the environment, workload, and resource availability.

Such heightened responsiveness is crucial for maintaining system stability and efficiency in the fast-paced and volatile landscapes of EoT-based systems. AI reduces the need for human intervention and probability of errors thereon, by making the system to make autonomous decisions regarding job scheduling and resource allocation. Autonomous decision-making is invaluable in sustaining system resilience and ensuring uninterrupted service delivery in EoT-based EMS. AI-driven approaches, therefore, not only ensure optimal resource utilization and cost reduction but also empower systems with predictive analytics and autonomous decision- making capabilities, paving the way for more resilient and efficient energy management solutions[88].

2.3.5.2 IoT and EoT Synergy

The next emerging trend to focus on is the synergy between IoT and EoT, with new studies exploring the integration of these technologies for advanced job scheduling and resource allocation. This enormously contributes to advancements in efficiency, adaptability, and real-time data processing. IoT and EoT work in conjunction to enhance real-time data processing, allowing for quicker response times and more informed decision-making processes in energy management. The implementation of IoT devices in tandem with EoT solutions facilitates immediate data analysis at the edge of the network, reducing latency and ensuring timely actions. The amalgamation of IoT and EoT plays a crucial role in optimizing energy consumption. IoT devices, coupled with EoT, enable the monitoring and control of energy usage in real-time, thus aiding in the development of energy-efficient solutions and the reduction of operational costs[86]. Moreover, this integration allows for the extraction of insightful information and patterns from vast data sets, empowering energy management systems to make more accurate and informed decisions. The integration of Internet of Things (IoT) and Edge of Things (EoT) in the field of energy management facilitates the creation of scalable and adaptable solutions that can effectively accommodate diverse workloads and environments. This ability to adapt is of utmost importance when it comes to effectively handling the diverse and evolving needs of modern energy infrastructures. Another major benefit being offered by this integration is that it enables the implementation of robust security measures and privacy-preserving techniques, safeguarding sensitive information and ensuring the integrity and confidentiality of data. In fact, this integration is a steppingstone in the advancement of intelligent, adaptable, and secure energy management solutions, addressing the ever-evolving demands of modern energy landscapes[74].

Thus, the literature reveals that job scheduling and resource allocation are fundamental components in the development of EoT-based EMS. While significant advancements have been made through optimization strategies, machine learning approaches, and the integration of emerging technologies, challenges such as complexity, scalability, and the dynamic nature of EoT environments necessitate continual research and innovation.

2.4 Optimization Techniques

Optimization techniques play a very important role in the further improvement of the effectiveness and efficiency of Energy Management Systems (EMS) in Edge-of-Things (EoT) environments. Optimization in Energy Management Systems (EMS) within the Edge-of- Things (EoT) environment is reliable for enhancing operational efficiency, reducing energy consumption, and improving sustainability[154]. Here are several optimization techniques that are prominently used in such settings:

2.4.1 Linear Programming (LP)

The first and the foremost technique to optimize the allocation of limited resources is Linear Programming (LP). By formulating linear equations, LP optimizes energy distribution and allocation in energy management systems, addressing problems related to energy scheduling and consumption. This method helps in achieving the best outcome in a mathematical model whose requirements are represented by linear relationships. The presented pseudocode provides a high-level outline of solving Linear Programming problems in Edge of Things, focusing on feasibility and optimality of the solutions. Specific implementations might involve variations of simplex method, Interior Point Methods, or other numerical optimization techniques, and could leverage distributed computing capabilities of Edge of Things for improvement in systems. It has been observed that adapting the methodology according to specific constraints, objective functions, and requirements inherent to the Edge of Things application in question is necessary for improvement[155].

2.4.2 Non-Linear Programming (NLP)

The presence of nonlinearity is a significant barrier as far as controlling the parameters in external environment are concerned. To deal with this, NLP has been identified as the most effective choice for controlling nonlinear systems, contributing to the improvement of system dependability and efficiency. NLP is used for optimizing the management and operation of renewable energy resources and is critical in handling energy conservation issues in non-linear energy systems. The specific method used to update the solution in the update solution function, and the way constraints are handled in the check feasibility function, should be meticulously designed, considering the specific problem context, available computational resources, and the requirement for real-time operation in the EoT environment. And thus, NLP becomes very useful for optimizing these complex systems where the objective function or the constraints are nonlinear[156].

2.4.3 Dynamic Programming (DP)

In addition to the previously employed methodologies, the Dynamic Programming (DP) also provides solutions to problems by breaking them into simpler subproblems, rendering it effectively for multi-stage decision-making in energy management. Dynamic Programming is implemented to determine optimal control strategies for energy storage systems, aiding decision-making processes in energy conservation. The purpose of this approach is to address the problems by breaking them into smaller and more manageable subproblems, and thereafter solving each subproblem only once while retaining their solutions. Dynamic Programming optimization could benefit from parallel computation for independent subproblems and efficient memory use for storing intermediate results and accommodating the resource-constrained nature of edge devices[157].

2.4.4 Genetic Algorithms (GA)

Genetic Algorithms (GA) find solutions to optimization problems based on the process of natural selection which is beneficial for optimizing complex systems in energy management. GA optimize energy consumption patterns and are instrumental in creating energy-efficient scheduling in smart grids. It contains search heuristics to find the exact or approximate solutions to optimization and search problems by mimicking the process of natural evolution[158].

2.4.5 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is another important optimization technique which is used for numerical optimization of problems in energy management systems, aiding in the optimization of power consumption and efficiency. As per facts, this method is developed by the social behavior patterns of organisms and is used for solving numeric optimization problems. Also, it is important to adapt and modify according to specific requirements and constraints of your Edge of Things environment[159].

2.4.6 Ant Colony Optimization (ACO)

Based on the foraging behavior of ants, an optimization technique called, Ant Colony Optimization (ACO) is applied to find the optimal paths in graph-based problems in energy systems. ACO is implemented to optimize routing paths and is able to improve the efficiency and reliability of energy distribution networks. Just like PSO, this technique also requires the need to adapt and modify the situation according to the specific requirements and constraints of your Edge of Things environment, taking the real-time and computational constraints into consideration[160].

2.4.7 Simulated Annealing (SA)

Simulated Annealing (SA) is a crucial technique employed for solving optimization problems in energy allocation and scheduling by exploring the solution space efficiently. It is a probabilistic technique used for approximating the global optimum of a given function, often used when the search space is discrete. SA is crucial for solving optimization problems in energy allocation and scheduling by exploring the solution space efficiently[161].

2.4.8 Greedy Algorithms

Greedy algorithms are also very popular as heuristic approaches for making locally optimal choices at each step in the hope of finding the global optimum in energy management systems. These algorithms are instrumental in solving problems related to resource allocation and energy distribution by making choices that seem the best at each step. The actual implementation and adaptation may vary based on specific use cases, system requirements, and constraints within the Edge of Things environment. So, it should be meticulously designed to consider the system's resource limitations and real-time processing needs. Also, a proper check of feasibility is important to ensure the efficiency and effectiveness of the algorithm in practical applications[162].

2.4.9 Constraint Programming (CP)

Constraint Programming (CP) is a technique implemented to optimize the scheduling and allocation of energy resources by resolving constraints effectively. The actual implementation may vary, and developers should consider the specific requirements and constraints of their Edge of Things environment. As per literature survey and application-based projects, constraint programming solves combinatorial problems by defining constraints and is pivotal in managing and optimizing energy systems. However, there is a need to focus on certain parameters such as timely and efficient processing of constraints when applying this optimization technique[163].

2.4.10 Machine Learning (ML) Based Optimization

Machine Learning (ML) based optimization models are the techniques used for data-driven decision-making processes and predictive analytics in energy management. ML models can not only predict energy consumption but can also optimize energy allocation by learning patterns and making informed decisions. As per the survey, machine learning models like neural networks, decision trees, and support vector machines are used for optimizing energy consumption and distribution in EMS. However, as per reports, there is a need for proper consideration of the resource constraints and real-time requirements of Edge of Things applications when applying ML-based optimization techniques[164].

2.4.11 Multi-Objective Optimization

This technique optimizes multiple conflicting objectives, offering solutions that balance different kinds of needs in energy management systems. It is applied to find trade-offs between conflicting objectives like cost, energy consumption, and emissions in energy system. This technique focuses on optimizing two or more conflicting objectives at a time, which is useful when there are trade-offs between different objectives[165].

2.4.12 Reinforcement Learning (RL)

Reinforcement Learning (RL) is a computational approach employed in decision-making processes, wherein an autonomous agent acquires knowledge through interactions with its environment. This technique holds particular relevance in the context of energy optimization, aiming to enhance the efficiency of energy consumption. RL agents optimize energy usage and allocation in real-time by learning the best actions to take in various states. An area of machine learning where an agent learns by interacting with its environment to achieve maximum cumulative reward is useful for making a sequence of decisions over time in EMS. This method is a basic representation of a Q-Learning approach in Reinforcement Learning and should be adapted according to the specific requirements, constraints, objectives, and environmental dynamics in Edge of Things systems[166].

2.4.13 Stochastic Optimization

Stochastic Optimization is a technique used to optimize decision-making under uncertainty, dealing with random fluctuations in energy supply and demand. This method has a very generic structure of a stochastic optimization algorithm in EoT. The algorithm should be adapted depending on the specific problem at hand (e.g., the objective function G, the feasible set X, the known distribution of work, etc.) and should be refined accordingly to ensure efficacy and reliability in the stochastic optimization process within the EoT environment[167].

2.4.14 Metaheuristic Algorithms

Metaheuristic algorithm provides high-quality solutions for optimization problems in energy management systems by exploring and exploiting the search space. These algorithms facilitate the optimization of energy scheduling and allocation by efficiently navigating through the solution space, ultimately leading to the identification of near-optimal solutions. High-level procedures are designed to find, generate, or select a heuristic that may provide a sufficiently good solution to an optimization problem. Metaheuristic algorithms are generally problem-independent, meaning they can be applied to a wide variety of optimization problems. However, the specific operations used to modify solutions (e.g., mutation and crossover in genetic algorithms) often need to be tailored to the specific problem being solved. This optimization method needs to be adapted based on the problem domain and the metaheuristic algorithm being utilized. This method always refers to specific algorithm guidelines and problem characteristics to refine the metaheuristic algorithm[168].

2.4.15 Cloud Computing-Based Optimization

Cloud Computing-Based Optimization enables scalable, flexible, and efficient solutions in energy management by handling large datasets and complex computations. It leverages the computational power of the cloud to perform optimization tasks, enabling more scalable and flexible solutions in EMS. Cloud computing offers scalable and powerful computational resources which can enhance the capability of EoT to solve complex problems, especially those that require significant computational power. The specific implementation details, including the structure of the solution “S” and the task “T”, and how they are processed in the cloud, would depend on the specific problem and optimization technique being used. It becomes highly important to ensure to comply with data privacy and security standards while offloading any task to the cloud, especially in EoT environments where data can be sensitive[169].

2.4.16 Distributed Optimization Algorithms

These algorithms enable optimization across distributed networks and are instrumental in scalable and flexible solutions in decentralized energy systems (Table 2.4). They optimize energy consumption across a network of devices and systems, facilitating coordination and management of distributed energy resources. These algorithms run on a network of computers, making them suitable especially for EoT environments where a central computational unit may not be available. This algorithm facilitates distributed optimization across multiple edge devices, emphasizing parallel local optimizations and collective decision-making based on the locally optimized solutions. The actual optimization strategy needs how solutions are represented, evaluated, and communicated and will depend heavily on the specific use case, optimization problem, and EoT environment characteristics[170].

Table 2.4: Various Optimization Algorithms

Optimization Technique	Application Suitability	Complexity	Scalability	Flexibility	Adaptability to Changes
Linear Programming (LP)	High for linear systems	Low	High	Medium	Low
Nonlinear Programming (NLP)	High for nonlinear systems	High	Medium	High	Medium
Dynamic Programming (DP)	High for multi-stage problems	Medium	High	High	Medium
Genetic Algorithms(GA)	High for complex, multi-modal problems	High	High	High	High
Particle Swarm Optimization (PSO)	High for numerical optimization problems	Medium	High	High	High
Ant Colony Optimization (ACO)	High for discrete optimization	Medium	High	High	High
Simulated Annealing(SA)	High for discrete search space	Medium	Medium	High	High
Greedy Algorithms	Medium for problems with optimal substructure	Low	Medium	Low	Low

It must be ensured by the user that the optimization strategy adheres to the constraints and requirements of the EoT system and respects privacy and security guidelines for data communication across devices[171].

From the literature survey, it has been observed that each of these optimization techniques has its own unique advantages in addressing specific challenges and requirements in Energy

Management Systems within Edge-of-Things environments. So, depending on the context and the problem at hand, one or a combination of these techniques can be deployed to enhance the efficiency and effectiveness of energy management strategies in EoT scenarios. The comparative table shows the role and responsibility of various optimization techniques in energy management systems. The optimal selection is context-dependent and may vary based on specific requirements, constraints, and objectives of the application. Based on the comparative study conducted, it is suggested that the most suitable algorithm be recommended for addressing application-specific requirements and challenges inside the workspace (Table 2.5).

Table 2.5: Comparative analysis of Optimization methods in relation to EoT-based EMS

Constraint Programming (CP)	High for constraint satisfaction problems	High	Medium	High	Medium
Machine Learning Based	High for data-driven problems	Varies	High	High	High
Multi-Objective Optimization	High for problems with conflicting objectives	High	High	High	High
Reinforcement Learning (RL)	High for sequential decision-making problems	High	High	High	High
Stochastic Optimization	High for problems with uncertainty	High	Medium	High	High
Metaheuristic Algorithms	High for a broad range of optimization problems	Varies	High	High	High
Cloud Computing-Based	High for scalable and flexible solutions	Varies	Very High	Very High	High

Distributed Optimization Algorithms	High for decentralized systems	High	Very High	High	Very High

Thus, it can be concluded that as far as Scalability and Flexibility are concerned, Distributed Optimization Algorithms and Cloud Computing-Based Optimization can be considered as the best as they can manage many variables and constraints and adapt to various problem structures. For Adaptability to Changes, Machine Learning-Based Optimization, Genetic Algorithms, and Reinforcement Learning are among the best as they can easily adapt to changes and uncertainties in the system. And for Complexity and Suitability, Linear Programming and Greedy Algorithms are simpler and more suitable for problems with a clear and optimal substructure.

The selection of the optimal technique should be dictated by the specific characteristics of the problem at hand, including the nature of the objective function, the type of constraints, the required level of accuracy, and the available computational resources. In many real-world applications, a hybrid approach combining the strengths of multiple techniques may yield the most effective and robust solution.

2.5 Algorithm for Classification

Classification of algorithms are integral in creating intelligent and efficient Energy Management Systems (EMS) within Edge-of-Things (EoT) environments. They are crucial for predicting and analyzing energy consumption patterns, leading to optimized energy use. When deploying classification algorithms on edge devices, it is crucial to consider the constraints of these devices, such as limited computational power, memory, and energy availability. As a result, lightweight and efficient algorithms are preferred for edge computing. There are key classification methods which are discussed below:

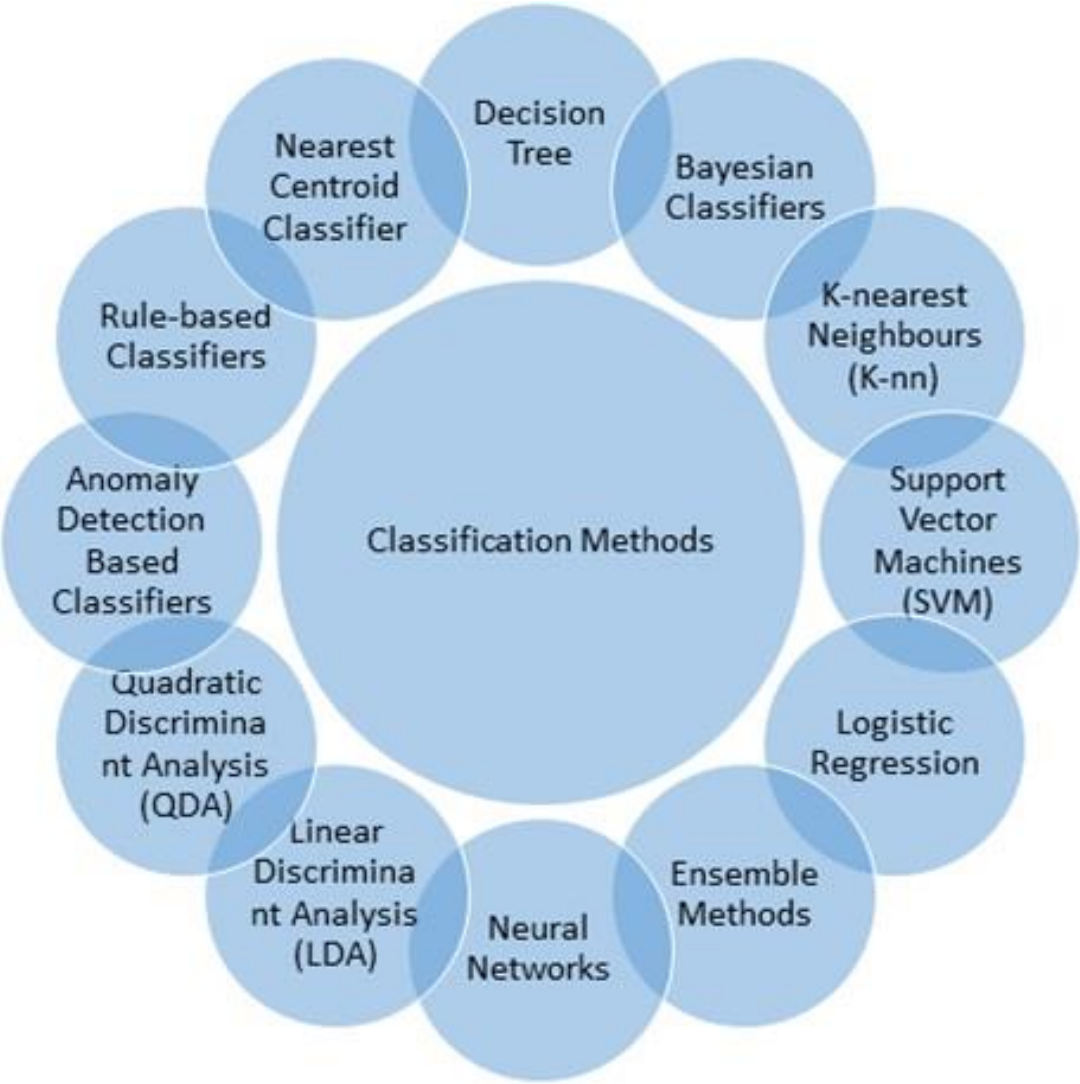

Figure 2.1 Classification Methods

2.5.1 Decision Trees

A decision tree is a versatile machine learning algorithm used for both classification and regression tasks. It mirrors our decision-making process, and its visual representation can be likened to an inverted tree, with the root at the top and branches expanding downwards, ending in leaves. It has been observed that the decision tree is built by partitioning data into subsets. Each internal node of the tree represents a “test” or “decision” on an attribute or feature of the data, each branch represents the outcome of that test, and each leaf node represents a class label. The process of learning a decision tree from data is recursive[172]. The dataset is continually split based on the best feature available, determined by a certain criterion. This process continues until one of the stopping conditions is met, such as the depth of the tree reaching a pre-defined limit.

2.5.2 Bayesian Classifiers

In the basket of classification methods, the Bayesian classifiers, particularly the Naive Bayes classifier, stand out due to their probabilistic foundation, simplicity, and surprisingly efficient performance in various tasks. Rooted in the Bayes' theorem, these classifiers offer a robust method to predict class membership probabilities. At the heart of Bayesian classifiers is Bayes' theorem, a principle in probability theory and statistics that describes the probability of an event based on prior knowledge. The theorem relates the conditional and marginal probabilities of events. The classifier calculates the posterior probability of a class given a set of features and then classifies the instance by picking the class with the highest posterior probability[173]. In practice, it involves computing the product of the individual probabilities of each feature occurring in a specific class and then scaling this by the prior probability of that class.

2.5.3 k-Nearest Neighbor's (k-NN)

The k-Nearest Neighbor's algorithm, commonly referred to as k-NN, is an essential example of instance-based and non-parametric learning in the domain of machine learning. Its core principle is rooted in the intuitive notion that similar data points, in each feature space, will often have similar outputs or labels. k-NN capitalizes on fast and reliable response to make predictions for new, unseen instances based on the labels of their neighboring data points[174].

2.5.4 Support Vector Machines (SVM)

Support Vector Machines (SVM) occupy a central position in the world of machine learning algorithms. Originally developed for binary classification, SVMs have been extended to handle multi-class classification, regression, and even outlier detection. At their core, SVMs aim to find the optimal hyperplane that best divides a dataset into classes. The underlying principle of SVM is to maximize the margin between two classes. In a two-dimensional space, this hyperplane can be thought of as a line, but in higher dimensions, it becomes a plane or even a hyperplane. The "support vectors" are the data points that are closest to this hyperplane, and they "support" or define the hyperplane, hence the name "Support Vector Machines"[175].

2.5.5 Logistic Regression

Logistic Regression, despite its name, is a foundational algorithm primarily used for binary classification tasks in the machine learning landscape. It estimates the probability that a given instance belongs to a particular category, thus making it a probabilistic statistical model. Stemming from linear regression, logistic regression's unique attribute is its utilization of the logistic function to squeeze the output of a linear equation between 0 and 1, making it suitable for estimating probabilities. The crux of logistic regression lies in its namesake: the logistic function, also known as the sigmoid function.

2.5.6 Ensemble Methods

Ensemble methods, a pivotal paradigm in the realm of machine learning, embody the principle that 'many heads are better than one.' Rather than relying on a single model's predictions, ensemble techniques leverage the collective power of multiple models to improve overall performance and reduce the chances of an erroneous prediction[176]. The central idea is that by combining multiple models, one can harness their individual strengths, offset their weaknesses, and, in the process, achieve better generalization and robustness. At a high level, ensemble methods involve training multiple models (often referred to as "base learners" or "weak learners") on a dataset and then devising a strategy to combine their predictions. The combination can be achieved either by using some form of averaging or voting for regression and classification problems, respectively.

2.5.7 Neural Networks

As per current research survey, modern artificial intelligence and deep learning lies the neural network, a computational model inspired by the way biological neural systems process information. Over the past decades, neural networks have significantly evolved, propelling advancements in diverse fields ranging from image recognition and natural language processing to game playing and medical diagnosis. A neural network consists of layers of interconnected nodes or "neurons." Each neuron receives inputs, processes them, and produces an output. The processing usually involves multiplying each input by a weight, summing up the weighted inputs, adding a bias, and then passing the result through a nonlinear activation function[177].

2.5.8 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a dimensionality reduction technique used primarily in the realms of statistics and machine learning. LDA's primary goal is to maximize the separability among known categories. Often used as a preprocessing step in machine learning pipelines, LDA can also function as a linear classifier. LDA operates under the assumption that the input variables are normally distributed and that each class has the same covariance matrix. The primary intuition behind LDA is to project data points from the original feature space into a lower-dimensional space in such a way that data points from different classes are as far apart as possible, while data points from the same class are as close as possible[178].

2.5.9 Quadratic Discriminant Analysis (QDA)

Quadratic Discriminant Analysis (QDA) is a classification technique that extends the principles of Linear Discriminant Analysis (LDA) while dropping some of its linear constraints. QDA is used for classifying instances into predefined groups or classes, but the decision boundary between these groups is quadratic rather than linear. The foundation of QDA, much like LDA, is grounded in Bayes' theorem. While LDA assumes that all classes share a common covariance matrix, QDA allows each class to have its own covariance matrix. This fundamental difference results in quadratic decision boundaries, making QDA more flexible than LDA in capturing relationships within data.

2.5.10 Anomaly Detection Based Classifiers

Anomaly detection, commonly known as outlier detection, is a technique used to identify patterns in data that do not conform to expected behavior. Such anomalies or outliers could be indicative of errors, fraud, system malfunctions, or even a new underlying pattern. When applied to classification tasks, anomaly detection-based classifiers often function by treating one class, typically the minority class, as the anomaly. Traditional classifiers operate under the premise that they have adequate examples from each class to learn the distinguishing characteristics. Conventional classifiers may struggle to recognize the minority class due to the vast imbalance in representation[179].

2.5.11 Rule-Based Classifiers

As per research communities, rule-based classifiers are a breed apart, relying not on complex mathematical models, but on a set of if-then rules to make classifications (Table 2.6). These rules are often interpretable and transparent, lending an intuitive feel to an otherwise complex decision-making process. A rule-based classifier makes decisions based on a set of rules. A rule is typically framed as "IF (condition) THEN (conclusion)." The "condition" part, often referred to as the antecedent or precondition, comprises one or more attributes or features. The "conclusion" part, known as the consequent, indicates the class label.

Table 2.6: Various Classification methods

Algorithm	Advantages	Limitations	Applications
Decision Trees	1. Simple to understand & interpret. 2. Can handle both numerical & categorical data. 3. Requires little data preparation.	1. Prone to overfitting. 2. Can be unstable due to small variations in data. 3. Often biased to classes with more levels.	1. Energy consumption prediction. 2. Fault detection in systems. 3. User behavior prediction. 4. Device management. 5. Network intrusion detection.
Bayesian Classifiers	1. Handles missing values. 2. Fast training. 3. Probabilistic approach offers confidence level.	1. Assumes independence of features. 2. Performance can be affected by irrelevant features. 3. Can be biased with imbalanced data.	1. Spam email filtering. 2. Sentiment analysis. 3. Predictive maintenance. 4. Weather forecasting. 5. Health monitoring.
k-NN	1. Simple & intuitive. 2. No training phase. 3. Adapts easily to changes.	1. Computationally intensive. 2. Sensitive to irrelevant features. 3. Requires meaningful 3. distance function.	1. Activity recognition. 2. Image recognition on edge devices. 3. Recommendation systems. 4. Gesture recognition. 5. Anomaly detection in networks.

2.5.12 Nearest Centroid Classifier

The Nearest Centroid Classifier (NCC) is a simple yet effective classification technique that is conceptually related to k-Nearest Neighbor's (k-NN) and Centroid-based clustering methods, such as k-Means. NCC focuses on the central point, or centroid, of each class in the training data to make its predictions. The foundational idea behind the Nearest Centroid Classifier is straightforward: compute the centroid for each class based on the training data, and for a new instance, predict the class whose centroid is closest to it. The centroid of a class is calculated as the average of all instances in that class. The comparative analysis of classification methods is given in Table 2.7.

Table 2.7: Comparative analysis of classification methods for energy management in relation to EoT

SVM	1. Effective in high dimensional spaces. 2. Uses a subset of training points. 3. Robust against overfitting.	1. Not suitable for large datasets. 2. Sensitive to noise. 3. Choice of kernel can be critical.	1. Image classification. 2. Handwriting recognition. 3. Bioinformatics applications. 4. Video surveillance. 5. Fault diagnosis.
Logistic Regression	1. Fast training and prediction. 2. Probabilistic results. 3. Can be regularized to avoid overfitting.	1. Assumes linearity between variables. 2. Can struggle with non-linear boundaries. 3. Sensitive to high correlation variables.	1. Predicting equipment failure. 2. User preference predictions. 3. Financial forecasting. 4. Traffic analysis. 5. Environmental monitoring.
Ensemble Methods	1. Boosts performance. 2. Reduces overfitting. 3. Handles missing data.	1. Can be computationally intensive. 2. More complex than individual models. 3. Choice of base models can impact performance.	1. Critical system monitoring. 2. Enhanced image recognition. 3. Data fusion from multiple sensors. 4. Robust speech recognition. 5. Advanced anomaly detection.
Neural Networks	1. Can model non-linear boundaries. 2. Adaptable to different tasks. 3. Can learn from raw data.	1. Requires large datasets. 2. Black-box nature. 3. Computationally intensive.	1. On-device speech recognition. 2. Real-time video analysis. 3. Smart home automation. 4. On-device language translation. 5. Advanced health monitoring.

LDA	1. Reduces dimensionality. 2. Assumes equal covariance for all classes. 3. Optimal for Gaussian distributed data.	1. Assumes linear boundaries. 2. Sensitive to outliers. 3. Assumes features are statistically independent.	1. Face recognition. 2. Biometric verification. 3. Pattern classification in signals. 4. Real-time motion sensing. 5. Environment pattern detection.
QDA	1. Can model quadratic boundaries. 2. More flexible than LDA. 3. Good for non-linear separable classes.	1. Requires estimation of more parameters than LDA. 2. Can overfit in high dimensions. 3. Assumes features are statistically independent.	1. Non-linear pattern recognition. 2. Advanced bio-signal classification. 3. Complex motion detection. 4. Adaptive user-interface designs. 5. Sound pattern analysis.
Anomaly Detection Based Classifiers	1. Detects new unseen patterns. 2. Works without labeled data. 3. Adaptable to changes in data patterns.	1. High false-alarm rate. 2. Requires good feature engineering. 3. Sensitive to data scaling.	1. Intrusion detection systems. 2. Fraud detection. 3. System health monitoring. 4. Fault detection in machinery. 5. Quality control in manufacturing.

Rule-Based Classifiers	1. Highly interpretable. 2. Can encode expert knowledge. 3. Adaptable to changing environments.	1. Can become very complex. 2. Can be sensitive to noise in data. 3. Might not capture all complexities.	1. Expert systems in critical operations. 2. Customizable user preferences. 3. Automated troubleshooting. 4. Personalized content delivery. 5. Environment-aware device management.
Nearest Centroid Classifier	1. Computationally efficient. 2. Simple and intuitive. 3. Scales well with large datasets.	1. Assumes homogeneous class distributions. 2. Sensitive to outliers. 3. Assumes linear class boundaries.	1. Quick initial classifications. 2. Scalable user segmentation. 3. Resource allocation based on device types. 4. Efficient initial image categorization. 5. Fast text classification for commands.

Hence, the literature survey has been done keeping all the associated factors into mind for understanding the gap in the concerned issue and for identification of tools and technology.

Based on literature, there are certain **Research Gaps** which are given below.

1. Sending enormous measure of information to the virtual computing platform causes hugeoverhead in terms of time, throughput, energy utilization, and cost

2. The cloud genuinely situated as extremely far away, so it is hard to achieve desired QoS aslatency and throughput.

3. Data centers are over-burdened to deal with huge amount of enormous information continuously and prompt confronting difficulties, i.e., capacity, security, and investigation

4. Distributed computing is difficult to oblige analytic engines for proficient preparation of enormous information.

2.6 Objectives

Keeping in mind the research gaps in the literature, the objectives of the study is given below:

1. To study and analyse the existing Energy efficient techniques in Edge-of things Ecosystem.
2. To design a framework for smart Energy Management system in Edge of things.
3. To develop an efficient energy scheduling algorithm using distributed learning for the proposed framework.
4. To validate and compare the proposed work with the existing techniques in the simulation environment

2.7 Conclusion

Edge of Things (EoT) technology with a focus on energy efficiency highlights the critical intersection of cutting-edge IoT developments and the imperative for sustainable energy use. This body of research underlines the importance of integrating energy-efficient practices into the fabric of EoT technology to not only enhance its operational efficiency but also to mitigate its environmental impact. By examining various methodologies, frameworks, and case studies, the survey sheds light on the innovative strategies being employed to optimize energy consumption without sacrificing performance. It reveals a growing recognition of the need for EoT devices and systems to be both technologically advanced and energy-conscious, pointing towards a future where energy efficiency is a cornerstone of technological development. This synthesis of EoT and energy efficiency research serves as a beacon for future investigations, urging continued exploration into how these technologies can evolve in harmony with environmental sustainability goals.

Further, in relation to objectives of the study, the research methodology is proposed in the third chapter and then the results and discussion will be given in chapter four. In chapter five, the conclusion and future scope of study will be discussed for further extension of the research work to be carried out by research communities and organizations for betterment in the EoT modules.

Chapter 3

Methodology

3.1 Introduction

As per recent reports, there has been a significant surge in the burden placed on cloud systems for data management and processing. As a result of which, users are facing considerable challenges in accessing services within the required time frames, particularly in relation to critical projects such as healthcare and monitoring. There are certain application areas, such as energy generation and distribution units, nuclear reactor control, Industrial tolls and process management, health care etc. where the users cannot tolerate the latency and processing of information. Also, it ensures zero tolerance for decision making by the experts in order to maintain strict control over the system. The existing literature indicates that technological advancements have made the edge of things slightly more advanced in addressing problems at local levels. However, there still a dearth in the development of strategies and planning for utilizing the edge of things module to facilitate prompt decision-making at this level. As per experts, there is need to understand the methodology for realization of edge of things concept in real-life scenario. The application area, situation and their interlinking of parameters is very important to build the logic for solution.

In the era of the internet of things, health care is such sector where there is need for quick monitoring decisions to save the life of human subjects, considering various drastic situations such as heart attacks, sudden spikes in heart rate, blood pressure, and abnormalities in other anthropogenic parameters[63]. It is very important to understand the EoT on local level for advanced research. Edge of Things (EoT) is one of the advanced level processing in relation to IoT. The term "Edge of Things" does not have a widely recognized or standardized technical meaning in the field of technology or IoT (Internet of Things). However, it is possible to interpret the term based on the components "Edge" and "Things" as they relate to IoT[180].

Edge: In IoT, "edge" often refers to edge computing. Edge computing involves processing

and analyzing data on or near the IoT devices or sensors themselves, rather than sending data to a centralized cloud server. This is done to reduce latency, improve real-time decision-making, save on bandwidth, enhance data privacy, and provide more efficient data processing[140].

Things: "Things" in IoT typically represent the various interconnected devices and sensors that make up the IoT ecosystem. These can include sensors, actuators, cameras, smart appliances, industrial machines, and more.

So, "**Edge of Things**" could conceptually refer to the intersection or integration of edge computing with the diverse array of IoT devices and sensors. It might imply a focus on how data is processed and managed at the edge of an IoT network, where devices interact with the physical world.

This study aims to highlight a significant medical concern related to patients/ senior citizens. It becomes highly imperative to closely observe the physical condition of human subjects to provide necessary care and rapid treatment to save their lives. In relation to the well-being of patients, there exists three primary tasks, such as:

1. Monitoring and observation of physiological signals
2. Making decisions for treatment
3. Cure ness of situation.

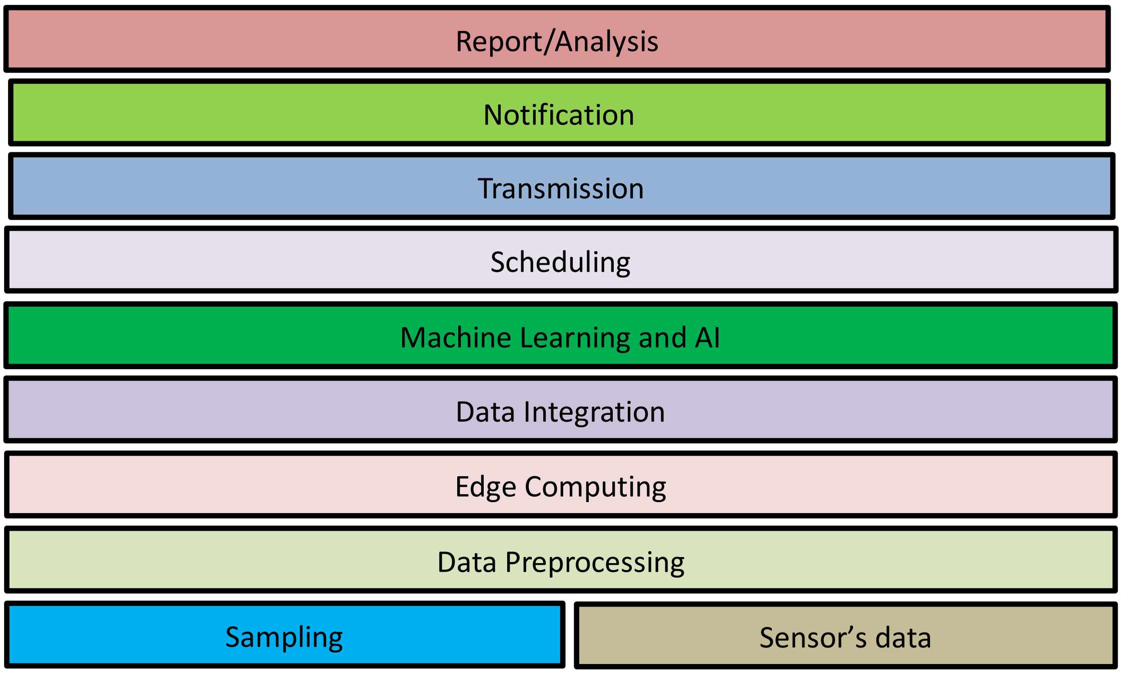

Figure 3.1: IoT-based Framework for Healthcare Monitoring

The figure 3.1 presents a hierarchical structure of an Internet of Things (IoT) framework designed for healthcare monitoring. The layers are organized in a vertical arrangement to represent the flow of data from the bottom to the top and the sequence of processing.
The data from the sensors was acquired at the fundamental level, where numerous health-related data points are gathered by IoT sensors.

i) Sampling layer is a critical concept in data analysis, focusing on the strategic selection of data points either at predetermined intervals or based on specific conditions. This method is pivotal in managing vast datasets, especially in scenarios like environmental monitoring or real-time analytics, where continuous data collection may not be feasible[181]. By collecting data at regular intervals, the approach ensures a manageable yet representative dataset over time. Alternatively, condition-based sampling targets specific events, significantly reducing unnecessary data accumulation. Once collected, this data undergoes further processing for various applications, including predictive analytics or anomaly detection. This selective data collection strategy optimizes resource use, balancing the need for comprehensive analysis with the constraints of processing capabilities and storage, thereby enhancing efficiency and sustainability in data-driven operations[182].

ii) Data Preprocessing layer encompasses the preliminary procedures involved in managing data, including the cleansing, normalization, and preparation of raw data for subsequent analysis.

iii) Edge Computing layer refers to the computational operations that occur near the data sources, with the aim of minimizing latency and alleviating the burden on central servers[183].

iv) Data Integration layer refers to the consolidation of data from several sources into a unified dataset, ensuring that all pertinent information is considered during analysis.

v) Machine Learning and AI layer involves the utilization of sophisticated algorithms and analytical models to analyze data, potentially offering predicted insights and assisting in decision-making.

vi) Scheduling layer is where the system's energy management strategy starts to demonstrate its effectiveness. The system functions conditionally, activating only the required sensors to collect further data when an abnormal health reading is detected, while keeping the remaining sensors inactive[184]. When all readings are within the usual range, most sensors transition into a sleep mode, resulting in a substantial decrease in energy consumption. The HRV sensor is the sole exception, as it remains operational to consistently observe the patient's condition. It offers crucial data on the autonomic nervous system, which is essential for promptly identifying potential health problems[185].

vii) **Transmission layer**, a protocol for managing energy is implemented. The primary function of this layer is to facilitate the seamless transmission of data, while also improving communication protocols to minimize energy consumption. The system gives precedence to time-sensitive data for quick transmission, while less critical information is transmitted less frequently[186]. The communication protocol adjusts according to the system's present power state and the importance of the information, guaranteeing minimal energy usage while maintaining the promptness and precision of vital health data[185].

viii) **Notification layer** facilitates the system's interaction with users, healthcare professionals, or automated systems by informing them about important events or necessary activities[187].

ix) **Report/Analysis layer**, involves the creation of reports, visualizations, and in-depth analysis that offer meaningful information to healthcare professionals and stakeholders, enabling them to act. The stacked picture underscores the significance of every layer in converting unprocessed sensor data into significant and actionable insights.

Keeping in mind the scenario, we have proposed a walking stick or band with Edge of Things capability including three primary tasks for advanced level care. This walking stick is one of the necessary amenities for an elderly person to freely move here and there. Based on the available studies, it has been observed that old-aged people may encounter significant challenges while engaging in walking activities, including instances of sudden falling, unexpected fluctuations in heart rate variability, and abrupt cessation of cardiac chamber function.

To tackle this situation, a very smart and diligent strategy is framed for the solution. A wireless module is proposed with thinking capability for real-time processing and observation of data and on the spot decision-making capabilities. This can be possible with use of artificial intelligence technology which can be readily integrated to sensor modules for critical thinking. Artificial Intelligence (AI) is one of the powerful tools for implementation of critical and logical thinking in machines for governing the desired tasks[188]. The artificial intelligence system is mimicking the human critical thinking approach to devise solutions to complex situations. Earlier, the AI tools were embedded on a central workstation for processing of data. For the data to reach the server, clouds were used, but due to enormous challenges such as latency, large data packet size, cost, and other factors, it gave boost to local level processing of data rather than central or remote area locations[86]. The AI tools are now readily available in the comfortable packages in the sensors and processing units for perfect decisions in relation to real time data processing. As per application and situation, effective planning and strategic implementation are essential so that the machine can do the needful with resource utilization.

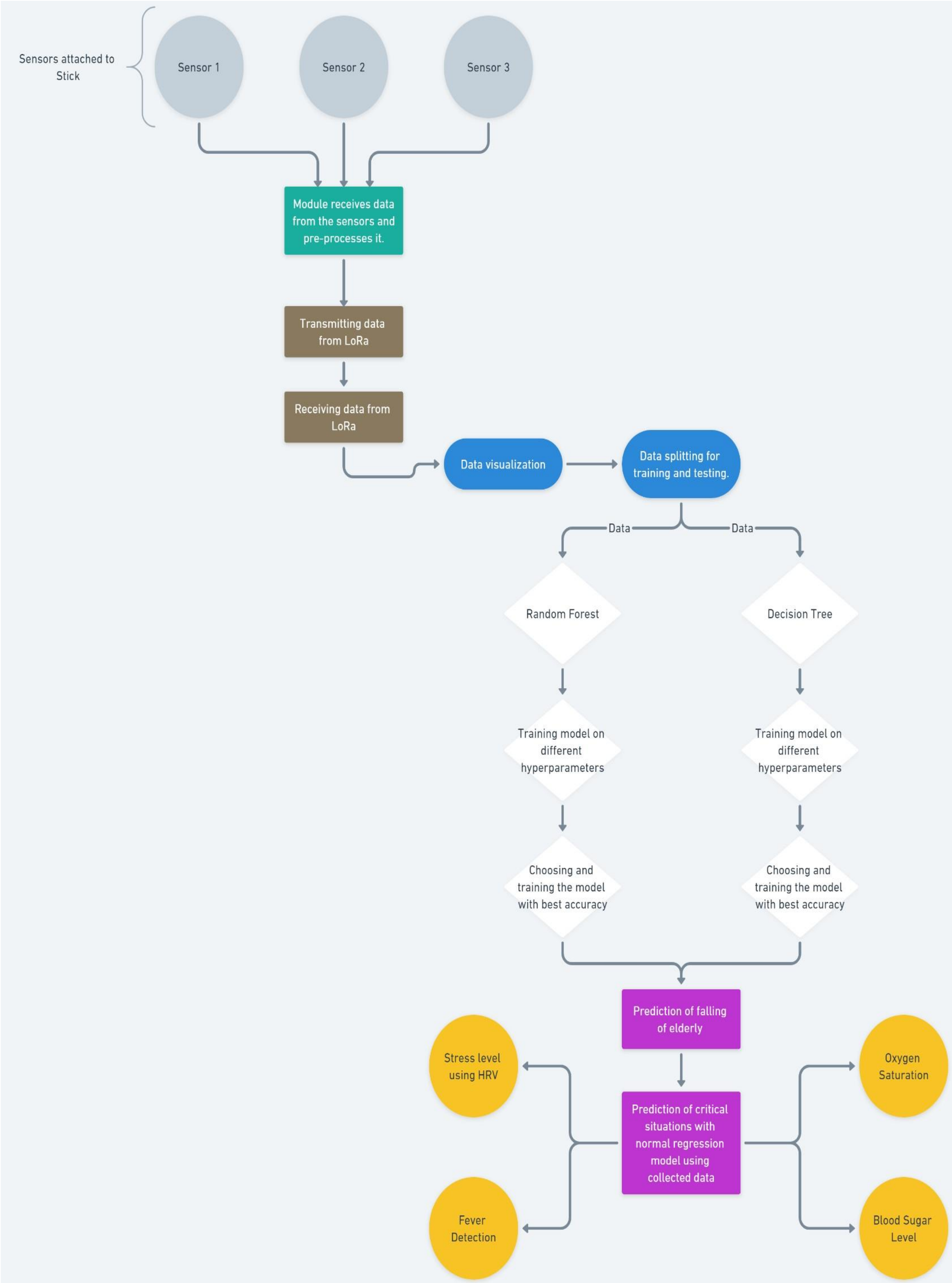

Figure 3.2: Flow chart for development of smart EoT based Waking stick for old age human subjects

The selection of workbench and modules needs rigorous analysis of available modules and tools for developing the smart application tools. The flow chart of proposed methodology is given in figure 3.2. The modules associated in the flow chart are explained below:

3.2 Sensor Array and database: Sensor is an electronic module that contains a transducer mechanism followed by a processing unit for generation of data with respect to the stimulation of the parameters to be measured. The fundamental objective of a transducer mechanism is to facilitate the conversion of energy from one form to another (preferably an electrical signal). The processing unit is one of the key sections where the signal is pre-processed to produce the desired data[189]. After transduction in a sensor, the signal typically undergoes several pre-processing steps to prepare it for further analysis. The heterogeneous process of sensing module is very important to understand and is given below:

1. **Buffering and Amplification**: Transducers, which convert one form of energy to another, often generate signals that are very weak for direct processing by subsequent sections. Buffering, provides an intermediary stage, which is liable for effectively preventing the weakening of these signals due to the input characteristics of later stages[190]. Subsequently, the amplification circuitry increases the amplitude of this signal, making it more conducive for further processing. These two steps are fundamental in ensuring the integrity and usability of the sensor's output.

2. **Filtering**: It has been observed that the real-world signals are often contaminated with unwanted frequencies or noise. So, the process of filtering refines these signals either by removing or attenuating the amplitude of the undesired components. Low- frequency drifts have the potential to induce a baseline shift, thereby masking the true signals. High-frequency noise can obscure details and cause false triggers in digital systems[191].

3. **Analog-to-Digital Conversion (ADC)**: As per demand of digital processing, if the sensor's output is analog, it needs to be converted into digital form. The ADC performs this operation through a two-step process. First is the process of sampling, which involves capturing the amplitude of an analog signal at specified intervals of time, and second is the process of quantization, which assigns a digital value to each sampled amplitude in the binary form of zero and one[192].

4. **Normalization**: Signals can vary in amplitude based on the conditions and the sensor's characteristics. Normalization scales the signal to a consistent range (often between 0 and 1), making it easier to compare and analyze, especially in digital systems.

5. **Baseline Correction**: Under different conditions, sensors might introduce an offset or drift in the signal. Baseline correction is a technique used to mitigate the influence of baseline shifts in a signal, thereby, ensuring that the signal remains centered around a 'zero' or baseline value.

6. **Calibration**: To ensure that a sensor's output truly represents the real-world value its measuring, regular calibration is necessary. The sensor's output may deviate from accurate values due to several factors such as environmental changes, sensor age, or other changes. Calibration thus aligns the sensor's output with known and recognized standards[193].

7. **Denoising, Rectification, and Feature Extraction**: In some advanced applications, further refinement of the signal is required. Denoising techniques aim to enhance the quality of a signal by effectively minimizing any residual noise[194]. Further, rectification (making all values positive) is often used in applications like electromyography, where the focus is on signal power or magnitude rather than its polarity. Feature extraction is essential in applications like machine learning, where specific characteristics or patterns in the signal are more important than the complete signal itself. So, the pre-processing of signals after transduction in sensors ensures that the raw data is refined, standardized, and made suitable for subsequent stages of processing or analysis[195]. Each of these phases plays a significant role in improving the quality, accuracy, and usefulness of the sensor's output.

In this study, the health parameters of human subjects, such as Pulse Rate, Heart Rate Variability (HRV), Blood Sugar level, Oxygen blood saturation (SpO2), Blood pressure level, were collected from Guru Nanak Charitable Trust, Jalandhar (Pb) for preparation of multi entity-based database. In conjunction with biomedical signals, the data of distance, Accelerometer data and body temperature was also measured with the use of concerned sensors. From the period of April 2022 to December 2023, the data set of more than 40,000 entities for each parameter was prepared for machine learning and testing of algorithms. It is mandatory to understand the signal types and ranges for development of control checks. The various biomedical signals in relation to patients under observation are discussed below:

3.2.1 Heart Rate measurement

Heart rate is the frequency at which the heart beats per minute (bpm). As per medical experts, it is a significant determinant of health in human subjects, especially old people. Measuring the pulse rate can provide valuable information about a person's cardiovascular health. The normal resting heart rate for older person varies from 60 to 100 bpm. Heart rate can be influenced by a multitude of factors, including activity level, fitness level, air temperature, body position, emotions, body size, and medication use. It is important to acknowledge that although the conventional range for a typical resting heart rate falls between 60 and 100 beats per minute (bpm), many healthy individuals have a resting heart rate outside of this range. If someone is concerned about their heart rate or if they notice sudden, unexplained changes, they should immediately seek medical advice[94]. As per research, there are various factors that are responsible for Pulse Rate variability in patients or old age, such as

- ❖ **Physical Activity**: Exercise or any physical exertion can increase the pulse rate.
- ❖ **Body Temperature**: Fever can elevate the heart rate.
- ❖ **Emotions**: Stress, anxiety, and excitement can raise the pulse rate.
- ❖ **Medications**: Some drugs can either elevate or lower the heart rate.
- ❖ **Age**: Typically, younger people have a faster resting heart rate than older individuals
- ❖ **Body Size**: Body size can influence heart rate, but the relationship is not straightforward.
- ❖ **Health Conditions**: Conditions like anemia, hyperthyroidism, and others can influence pulse rate.

Pulse rate is measured by an Optical Infrared sensor. The primary mechanism of a tiny pulse meter sensor is photoplethysmography (PPG). PPG sensors operate by emitting light (usually green LEDs, but other wavelengths can also be used) onto the skin. The amount of light that the blood absorbs changes as it circulates through the vessels. The light that is not absorbed is reflected by the photodetectors in the sensor. By measuring these changes in reflected light, the sensor can determine the pulsatile variations caused by the heart's pumping action, allowing it to calculate heart rate[109]. Figure 3.3 shows the MIR’s multifunctional Pulse oximeter with SpO2 measuring device:

Figure 3.3: MIR's Pulse oximeter with SpO_2 measuring sensor

3.2.2 Blood sugar level measurement

Blood sugar levels in patients under observation can be influenced by a range of factors that differ from those affecting younger individuals. Sometimes, the medical disbalance can lead to changes in glucose metabolism, and patients may have a higher prevalence of type 2 diabetes or other metabolic disorders. Understanding and managing blood sugar levels in this population is crucial for their overall health and well-being. It has been observed that the insulin resistance can increase in patients, even in those without diabetes. This implies that the human body may exhibit reduced responsiveness to insulin, resulting in elevated levels of glucose in the bloodstream[196]. The risk of developing type 2 diabetes thus increases with medical condition/ age.

Factors affecting blood sugar levels in older adults:

❖ **Medications:** Patients often take multiple medications for various conditions. Some of these medications might impact blood glucose levels, either increasing or decreasing them.

❖ **Coexisting Health Conditions**: Conditions like kidney disease, thyroid disorders, or liver issues can also influence blood sugar control. Additionally, cognitive decline or dementia can impact an individual's ability to manage their diabetes effectively.

- **Decreased Physical Activity**: Reduced mobility or physical activity can lead to increased insulin resistance, making it harder to control blood sugar levels.

- **Dietary Changes**: Patients might have altered diets due to difficulties in chewing, digestion issues, or other age-related factors, which can influence blood sugar levels.

- **Weight Changes**: Unintended weight loss can be common in patients, which might impact blood sugar levels and medication requirements.

Glucose, commonly referred to as blood sugar, plays a vital role in numerous cellular processes, with the brain being particularly reliant on glucose as its principal source of energy. Hypoglycemia, characterized by abnormally low blood sugar levels, poses a significant risk to an individual's life if not properly addressed. The specific threshold for hypoglycemia can vary among different health organizations and can be influenced by contextual factors such as the treatment of diabetes[109]. However, hypoglycemia is generally defined as a blood sugar level below 70 mg/dL (3.9 mmol/L). The symptoms and severity of hypoglycemia can vary based on how low the glucose level drops. Mild Hypoglycemia typically starts at blood sugar levels slightly below 70 mg/dL (3.9 mmol/L). The Symptoms includes Trembling or shaking, Sweating, Hunger or Palpitations. In the care of moderate Hypoglycemia, the levels continue to drop, additional symptoms might manifest such as Mood changes, like irritability, Fatigue, Blurred vision, Difficulty concentrating at work. The case is severe hypoglycemia in which the level of blood sugar falls below 40 mg/dL (2.2 mmol/L) but can vary among individuals. At this stage, neurological symptoms become pronounced due to the brain's decreased glucose supply, which includes confusion or disorientation, Seizures, Loss of consciousness or coma[196]. As per medical practitioners, severe hypoglycemia in the absence of prompt treatment can lead to permanent brain damage or death. So, it is essential to understand that, while these are general thresholds, the exact level at which someone might experience symptoms can vary. Some individuals, especially those with a history of recurrent hypoglycemia or long-standing diabetes, may have "hypoglycemia unawareness" where typical symptoms are blunted, and severe symptoms can arise without much warning.

3.2.3 Oxygen blood saturation (SpO_2)

SpO2 represents the percentage of hemoglobin binding sites in the bloodstream occupied by oxygen. It is an essential metric in assessing the oxygenation status of patients. In a healthy individual, normal oxygen saturation levels usually range from 95% to 100%. The optical pulse oximeter is used to measure the oxygen blood level[63]. In patients, several factors and age-related physiological changes can impact oxygen saturation such as

❖ **Respiratory System Changes**: As per condition of patients, there will be a decrement in lung elasticity, diaphragmatic strength, and thoracic cage flexibility. Additionally, the number of functional alveoli, the tiny air sacs are responsible for gas exchange, decreases. These changes can reduce overall lung function and oxygen exchange efficiency.

❖ **Decreased Cardiac Efficiency**: It has been observed by the medical practitioners that the age-related changes in the cardiovascular system, such as reduced cardiac output, can impact the ability to circulate oxygen-rich blood efficiently throughout the body[187].

❖ **Diseases and Co-morbidities**: Sometimes, the patients are more susceptible to chronic diseases like Chronic Obstructive Pulmonary Disease (COPD), heart failure, pneumonia, and other respiratory infections, which can compromise oxygen saturation levels.

❖ **Decreased Activity Levels**: Reduced mobility and activity can lead to decreased respiratory muscle strength and overall respiratory function.

❖ **Medications**: Some medications are common in patients under medical observation, like certain sedatives can suppress respiratory drive and impact oxygenation.

❖ **Sleep Disorders**: Conditions like sleep apnea, more prevalent in older adults, can lead to intermittent drops in oxygen levels during sleep.

❖ **Sarcopenia**: This refers to the loss of muscle mass with aging, which can also involve respiratory muscles, further affecting breathing and oxygenation.

It is imperative to acknowledge that age-related physiological changes may have an impact on respiratory function and oxygen saturation levels. However, it is crucial to seek medical assessment if an individual consistently exhibits a SpO_2 reading below 95%. Hypoxemia, a condition characterized by reduced oxygen saturation, can lead to symptoms like shortness of breath, confusion, cyanosis (bluish discoloration of the skin), and increased heart rate. Chronic hypoxemia can have detrimental effects on organs and overall health. Regular check-ups, pulmonary function tests, and other diagnostic evaluations can help in the early detection and management of respiratory issues[196]. Proper management of chronic diseases, avoiding smoking, staying active, and getting vaccinations (like the flu and pneumonia vaccines) are strategies to help maintain better lung health in old age.

3.2.4 Blood pressure level

Blood pressure (BP) is a critical vital sign that measures the force of blood against the walls of the arteries as the heart pumps it around the body. As per medical condition, the arterial walls tend to become stiffer, which can lead to increased blood pressure. Nevertheless, it is worth noting that patients' individuals are also susceptible to low blood pressure, particularly a condition known as postural hypotension[63]. This condition entails a decline in blood pressure especially when standing up from a sitting or lying position. Here are general blood pressure categories as outlined by the American College of Cardiology (ACC) and the American Heart Association (AHA). The range of normal blood pressure varies from Systolic (upper number): Less than 120 mm Hg to Diastolic (lower number): Less than 80 mm Hg. In case of elevated blood pressure, the range varies from Systolic: 120-129 mm Hg to Diastolic: Less than 80 mm Hg. Similarly, in the case of Hypertension Stage, the range of blood pressure varies from Systolic: 130-139 mm Hg to Diastolic: 80-89 mm Hg. In worst conditions, such as Hypertensive Crisis, the range of blood pressure varies from Systolic: Over 180 mm Hg to Diastolic: Over 120 mm Hg.

From the research findings, it has been observed that patients who have hypertension and are at a paramount risk of cardiovascular disease, as determined by other risk factors or pre-existing disorders, are typically advised to maintain a blood pressure target below 130/80 mm Hg. Likewise, those who fall within the age group of 65 years and beyond, and who possess a lower risk profile or have concerns about the tolerability of medication, may find it suitable to choose a more conservative blood pressure target, such as maintaining levels below 140/90 mm Hg. It is essential to recognize that these are general guidelines, and individual targets should be set in collaboration with a healthcare provider. Factors that might influence the ideal blood pressure target in patients include overall health status, the presence of other medical conditions, the risk of side effects from antihypertensive medications, and life expectancy[110]. It has been observed from the patients, that the indications of hypotension, such as vertigo or a sensation of faintness, particularly upon standing. The presence of postural (or orthostatic) hypotension, necessitating the implementation of an appropriate therapeutic strategy.

3.2.5 Distance Sensor

The distance sensor, which is also known as a range sensor or proximity sensor, is a device that measures the distance or proximity to an object without making physical contact with it. These sensors can operate over various ranges, from a few millimeters to several meters, and their applications span a wide variety of domains, from industrial automation to robotics, automotive safety systems, and everyday consumer electronics. There are various physical mechanisms by which the distance sensor works. Each method has various parameters in terms of instrumentation, cost, and power consumption (depending on the project)[197]. Electronic distance sensors are widely used to measure the distance to an object using electronic principles such as reflection and refraction of light. Different types of electronic distance sensors employ different techniques and principles to gauge distances. The methods are discussed below.

1. **Ultrasonic Distance Sensor**

These sensors work based on the reflection of ultrasonic sound waves. The sensor is equipped with both a transmitting component and a receiving component. The transmitter emits an ultrasonic sound wave. This sound wave travels through the air and hits an object. After striking the object, the sound wave is reflected and captured by the receiver[110]. The time taken for the wave to propagate to the object and return is computed to obtain the measurement.

2. Infrared (IR) Proximity Sensor

These sensors operate using infrared light, relying on either reflection or interruption of the light. An IR LED emits infrared light. When employing a reflection-based technique, the incident light undergoes reflection when encountering an object, subsequently gets detected by an infrared (IR) photodiode or phototransistor[198]. The magnitude of the received light can serve as an indicator of distance, as objects in closer proximity tend to exhibit a higher degree of light reflection.

3. Laser Distance Sensor (Time-of-Flight)

In this sensor module, a laser is used to measure the time taken by light to travel to an object, its subsequent reflection, and then its subsequent return. A laser diode emits a beam of light towards the target. The light hits the target and thereafter reflects on the receiver of the sensor. The photodetector is responsible for capturing the light that has been reflected. By measuring the time, it takes for the light to travel and return (the "time of flight"), the sensor can determine the distance. This is because light travels at a constant speed[199].

4. Capacitive Proximity Sensor

The presence of an object can change the capacitance of a sensor's electric field. An oscillating circuit generates an electric field. When an object approaches or contacts the sensor, it modifies the capacitance of the field[187]. This change is identified and subsequently analyzed to ascertain the object's presence and its potential distance.

5. Inductive Proximity Sensor

This sensor detects metallic objects based on changes in an electromagnetic field. The sensor has a coil that generates an electromagnetic field. When a metallic object approaches the sensor, it induces eddy currents in the object. These eddy currents alter the sensor's original electromagnetic field. The change in the field can be detected, signaling the presence of the metallic object. The unprocessed and raw data obtained from various sensors is often subjected to internal electronic circuitry for the purpose of converting it into a more practical and meaningful format[63]. This can be a digital signal, an analog voltage proportional to the distance, or even a specific value read out on a display. The exact nature of the output depends on the sensor's design and its intended application. The distance sensor used in this study is HC-SR04 ultra-low power ultrasonic sensor shown in figure 3.4. This sensor has three consecutive steps to measure the distance reading in meter. The range of this sensor varies from 80 to 90 meters.

i) **Emission & Reception**: Most distance sensors work on a simple principle where they emit some form of energy (e.g., sound waves, light, or electromagnetic fields) and then measure the energy that is reflected or the interruption of this energy.

ii) **Distance Calculation**: Once the reflected energy is received, the sensor processes this data to calculate the distance. This is often done by measuring the time between emission and reception (as in ultrasonic or laser time-of-flight sensors) or by gauging the intensity of the returned signal[196].

iii) **Output**: Once the distance is calculated, the sensor outputs this data in a form that can be read by other devices or controllers, such as a voltage, current, digital signal, or even a direct readout.

Figure 3.4: The ultrasonic sensor for distance measurement in meters

3.2.6 Accelerometer

An accelerometer is an electronic device that measures proper acceleration, which is the acceleration experienced relative to free fall. An electronic accelerometer translates mechanical motion into an electronic signal that systems other electronic systems can process and understand. It can directly measure any accelerative forces applied to the object (it is attached). Accelerometers are commonly utilized in electronic systems, particularly in the form of microelectromechanical systems (MEMS). These devices have gained extensive popularity and are extensively employed in a range of applications such as smartphones, fitness trackers, automotive systems, and several other domains[109]. Most MEMS accelerometers are made using silicon micro-machining technology. They have a mass (or "proof mass") suspended by tiny beams inside a small chip. This mass is free to move (to a certain extent) within the chip when accelerative forces act upon it. When the accelerometer experiences an acceleration, due to Newton's second law, the proof mass inside exhibits a tendency to maintain its position while the rest of the chip moves causing the mass to displace relative to its immediate surroundings. This displacement due to external acceleration causes stresses in the tiny beams that suspend the mass[94]. There are various electronic mechanisms to measure acceleration:

i) **Capacitive Sensing**

This is the most common method used in MEMS accelerometers. The proof mass and its surrounding structure form a set of capacitors. As the mass moves due to acceleration, the distance between the plates of these capacitors' changes, causing a change in capacitance. By measuring this change in capacitance, the displacement (and therefore the acceleration) can be determined[63].

ii) Piezoresistive Sensing

The small beams or structures that suspend the proof mass have piezoresistive properties, meaning their resistance changes when they are mechanically deformed. As the mass moves and causes stress in these beams, the change in their resistance is measured, which can then be correlated to the acceleration. There are other, less common sensing mechanisms like piezoelectric or thermal bubble-based accelerometers. The raw signals obtained from the sensing mechanisms, such as variations in capacitance or resistance, tend to be rather low in magnitude. Therefore, to enhance their strength and facilitate further analysis, these signals are amplified and subjected to processing either through on-chip or external circuitry. This processing can involve analog-to-digital conversion, noise filtering, and other signal conditioning to produce a usable output. Most basic accelerometers can measure acceleration in one direction or axis. However, many modern devices combine multiple accelerometers to measure acceleration in two (X and Y) or three (X, Y, and Z) axes. The application includes certain areas such as Motion Detection, Gesture Recognition, Vehicle Dynamics, Fitness Tracking, wearable devices to track steps and other activities[187].

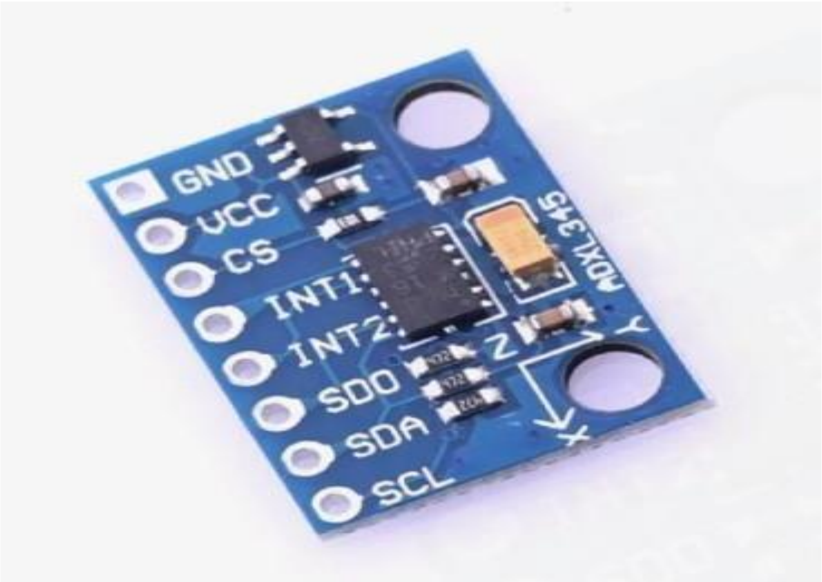

Figure 3.5: ADXL345 3-Axis Digital Accelerometer Sensor

In this study, ADXL345 3-Axis Digital Accelerometer Sensor (Figure 3.5) is used to measure the data.

3.2.7 Wireless Module (LoRA based Transmitter and Receiver)

LoRa (Long Range) is a modulation technique and a protocol for wireless communication that allows long-range data transmission with low power consumption. It is particularly popular for Internet of Things (IoT) applications due to its ability to support long-range communications while using minimal power. LoRa technology operates within the sub-gigahertz frequency range, exhibiting a distinctive modulation technique that confers resistance to interference and enables long-range transmission capabilities[198].

LoRa Transmitter

A LoRa transmitter is a device or module that sends out data using the LoRa modulation technique. It encodes and transmits the data over the radio frequency (RF) spectrum using LoRa's unique spread spectrum modulation. This modulation spreads the information across a wide frequency bandwidth, which makes the transmission more resistant to noise, interference, and fading. Its functionality includes Encoding the data for transmission, Modulating the data using the LoRa spread spectrum technique, Transmitting the modulated data over the RF spectrum[200].

LoRa Receiver

A LoRa receiver is a device or module designed to detect and decode data transmitted by a LoRa transmitter. The demodulation process of the LoRa receiver is essential for precisely retrieving the data, as it operates within the same frequency bandwidth utilized during the transmission, owing to the spread spectrum characteristics of LoRa. Its functionality involves the process of capturing RF signals from the surrounding environment, demodulating the LoRa spread spectrum signal to extract the original data, and subsequently decoding and transmitting this retrieved data to subsequent systems or applications[201]. In this study, LoRA 915MHz Shield-TTGO T-SIM7000G Module is used for establishment of Transceiver communication (Figure 3.6). The specification of LoRA module is given below:

1. Modulation Technique

TTGO T-SIM7000G LoRa uses a modulation technique called Chirp Spread Spectrum (CSS), which is a derivative of spread spectrum modulation. CSS improves signal robustness by reducing the impact of interference, hence facilitating more accurate detection, even in the presence of noise on the communication channels. This modulation technique allows LoRa to provide a trade-off between range and data rate. The LoRa technology has the capability to dynamically adjust its data rates by changing the spreading factor (SF), which serves as a key component inside the CSS modulation[198]. Higher spreading factors increase the time on air and range but reduce the data rate, and vice versa.

2. Frequency Bands

LoRa typically operates in license-free sub-gigahertz frequency bands, like 868 MHz in Europe and 915 MHz in North America. These bands are less crowded than the 2.4 GHz band, which is widely used by Wi-Fi, Bluetooth.

3. Long Range

LoRa can achieve extremely long ranges, often several kilometers in urban areas and tens of kilometer's in less dense areas. The long range is achieved by its robust modulation technique and the ability to detect very weak signals.

Figure 3.6: LoRA Transceiver module

4. Low Power

TTGO T-SIM7000G LoRa devices have been specifically designed to optimize energy efficiency. Depending on the specific application and frequency of usage, these devices have the capability to operate on small batteries for extended periods of time. This makes them ideal for remote sensors or devices that are challenging to access and service regularly

.

5. Adaptive Data Rates

LoRa devices can adjust their data rates based on the signal quality and range. This adaptability ensures a balance between communication speed and range, optimizing for the current conditions.

6. Network Structure

In TTGO T-SIM7000G-Lora module, a typical Lora WAN network is used which supports multiple end-node devices to communicate with gateways. These gateways are connected to a network server that manages the network[202]. The data collected by devices is aggregated by the gateway and transmitted to a centralized system, such as a cloud server, for the purpose of analysis and computation.

7. Security:

Lora WAN includes built-in encryption. It uses AES-128 encryption to ensure data security, device identity, and network integrity.

8. Collaborative Channels

LoRa can use multiple channels simultaneously, increasing the system's capacity. If a channel becomes congested or noisy, LoRa devices can switch to a cleaner channel. So, the combination of the Chirp Spread Spectrum modulation, adaptive data rates, low power design, and multi-channel capability allows LoRa to offer long-range, energy-efficient wireless communication for various IoT applications[198]. Whether its agricultural sensors spread across vast farms, water meters in city infrastructure, or tracking devices on wildlife, LoRa offers a compelling solution for situations where range and battery life are paramount.

3.3 Machine Learning based Algorithms

As shown in figure 3.1, the data of sensors is communicated by Lora transceivers to load the data on local processing units. The local processing unit contains data validation and machine learning based algorithms to effectively process the information according to conditionals traps and then generates output based on the hierarchy and rank of classified parametric information. Figure 3.7 depicts the flowchart illustrating the process of machine learning based processing. Machine learning (ML) is a subfield of artificial intelligence (AI) focused on the development of algorithms that allow computers to learn from and make decisions or predictions based on data. Instead of being explicitly programmed to perform a task, a machine learning model uses patterns in data to make informed decisions[133]. Research reports have indicated various classification methods. However, upon comparison, it has been found that the random forest, decision tree and Support Vector Machine algorithms are the methods which are widely and extensively employed due to their distinct qualities and as per nature of data collected in this study.

A Decision Tree is a flowchart-like structure where each internal node represents a feature, the branch signifies a decision rule, and each leaf node indicates an outcome. Decision Trees, although conceptually clear and visually intuitive, frequently exhibit a tendency to overfit, particularly when they possess a significant depth thereby leading to poor generalization on new data[164].

On the other hand, a Random Forest is an ensemble method that generates a 'forest' of multiple decision trees. Instead of relying on a single tree, it aggregates the predictions of numerous trees, each trained on a random subset of the data[203]. This process typically makes Random Forests more accurate and less susceptible to overfitting compared to a single Decision Tree.

A Support Vector Machine (SVM) is a powerful and versatile supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best divides a dataset into classes. The core principle of SVM is to identify the optimal separating hyperplane which maximizes the margin between different classes. Data points closest to the hyperplane on either side, known as support vectors, are crucial in defining the hyperplane and thus the decision boundary. SVMs are particularly effective in high-dimensional spaces and are versatile as they can accommodate different types of kernel functions to handle non-linearly separable data[175]. However, choosing the right kernel and tuning parameters like the penalty parameter C and the kernel-specific parameters can be complex, and SVMs can become less effective if the data is very noisy or the number of features far exceeds the number of samples, potentially leading to overfitting.

However, the trade-off is increased complexity, as visualizing or interpreting the entire forest becomes impractical. Furthermore, it should be noted that Decision Trees exhibit determinism, since they consistently generate the same structure when applied to a specific dataset. On the other hand, Random Forests and SVM possess an inherent unpredictability that can lead to variations in their structure, unless the random seed is set to a fixed value. Despite their added complexity, the robustness and superior performance, Random Forests are often preferred choice in various applications over Decision Trees algorithm and SVM due to best space time complexity than other algorithms[203].

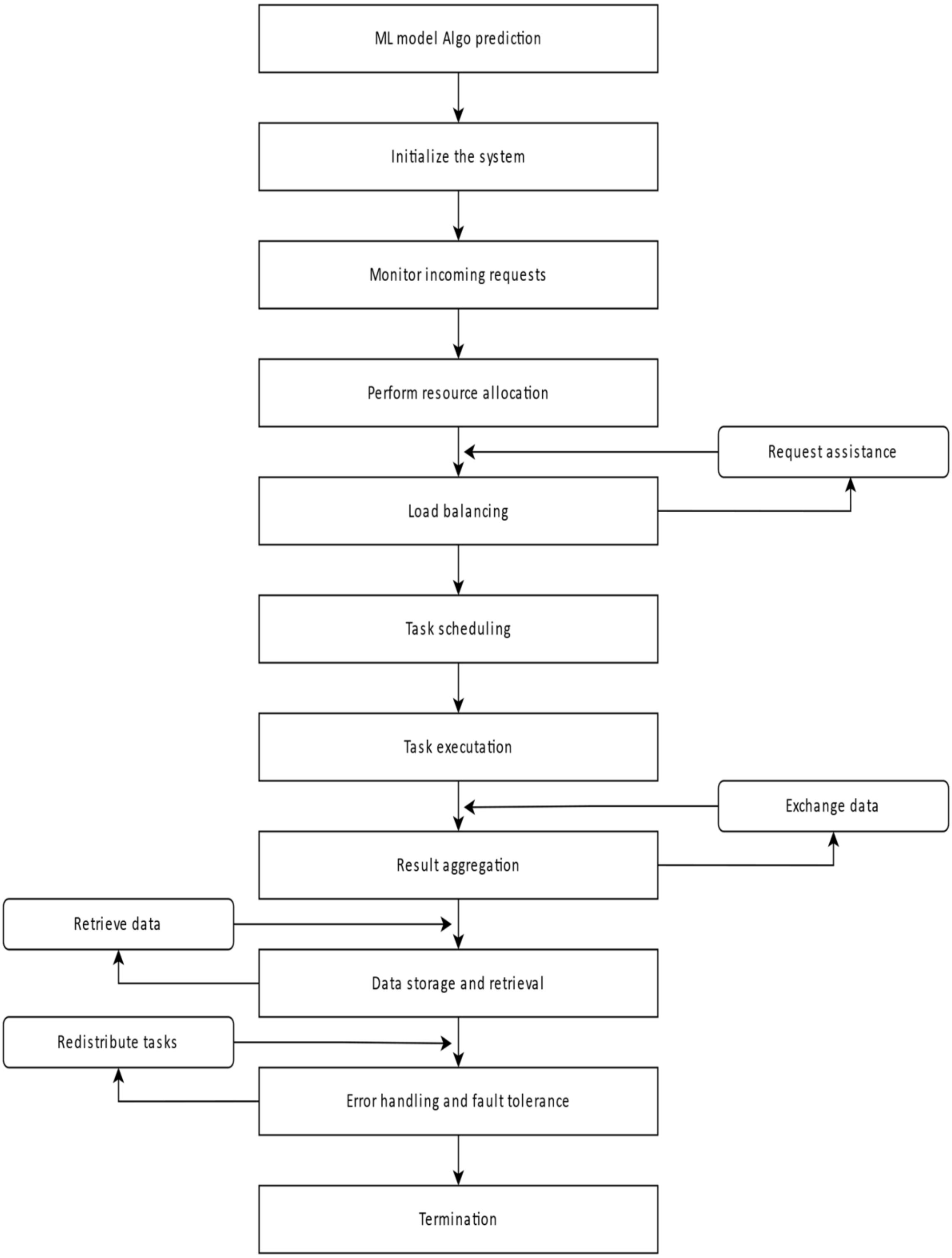

Figure 3.7: Flow chart of Machine learning based algorithm for task scheduling and execution

In this study the following algorithm steps are adopted to initialize the energy efficient EoT application in relation to local server-based computing.

Algorithm 1 System Operation Algorithm

```
Initialize the system
if (Monitor incoming requests) == True then
    Perform resource allocation
else
    Terminate
end if
if (Allocation) == Done then
    Perform load balancing
    Perform task scheduling
    Execute tasks
end if
Aggregate results
Store and retrieve data
if (Any error occurs) != NULL then
    Handle errors and ensure fault tolerance
    Terminate the system
end if
```

The received data is further processed by random forest, decision tree and SVM classification methods. The algorithm of these classification methods is implemented on local server. The algorithm works by building many decision trees, each of which is trained on a different subset of the data using a random subset of features. During training, each decision tree learns to make a prediction by recursively splitting the data into smaller subsets based on the selected features and their values. The process of splitting persists until a predetermined stopping criterion is met, which may include reaching the maximum depth of the tree or having a minimum number of samples necessary to perform a split[204]. Once, the algorithms were trained, the processing time and accuracy will be calculated. After that based on the threshold of accuracy and relative time consumption conditions, the best algorithms will be selected to make predictions on new data. The decision tree algorithm predicts the output value based on the input features, and the final output is determined by taking the majority vote of all the predictions.

As per literature, one of the benefits associated with employing a random forest model is its reduced susceptibility to overfitting in comparison to a other classifiers. The reason for this is that the model averages the predictions of multiple trees, which serves to mitigate the variability in the predictions, thereby enhancing the model's overall performance.

Furthermore, the utilization of random feature selection throughout the training process enhances the model's resilience to noise present in the input data. Random forest models are widely used in a variety of machine learning tasks, such as predicting customer churn, detecting fraudulent transactions, and classifying images. They are a powerful and versatile algorithm that can handle both numerical and categorical data and can be easily parallelized to handle large datasets. The max_depth parameter controls the maximum depth of each decision tree in the random forest. It helps to prevent overfitting by limiting the number of splits each tree can make. A higher max_depth may result in a more complex model, but it may also increase the risk of overfitting. The parameter of the estimator determines the quantity of decision trees to be included in the random forest[203].

A higher number of trees may improve the accuracy of the model, but it may also increase the computational cost and slow down the training process. Both hyperparameters should be chosen carefully to balance model complexity, accuracy, and computational resources. In general, increasing max_depth and estimators may lead to better performance, but it is important to avoid overfitting and to consider the trade-off between accuracy and computational cost. A decision tree model is a type of supervised learning algorithm that is used for classification and regression tasks. It works by dividing the input data into smaller subsets based on the values of selected features. The decision tree consists of nodes that represent features, branches that represent decisions, and leaves that represent the outcome or class label.

During training, the algorithm selects the best feature to split the data based on a criterion such as information gain. The chosen attribute is employed to generate a new node inside the tree, wherein the dataset is divided into smaller subsets based on the values associated with said attribute[203]. This process is repeated recursively for each subset until a stopping criterion is met, such as a maximum depth or a minimum number of samples per leaf. At this point, the tree is complete, and the final class label is assigned to each leaf based on the majority vote of the samples in that leaf. During prediction, the input data is passed down the tree, and each decision node makes a binary decision based on the value of the corresponding feature. The prediction is made at the leaf node based on the majority class label of the samples in that leaf. Decision trees have gained wide popularity due to their interpretability and ability to handle both categorical and numerical data. Nevertheless, the decision tree model is susceptible to overfitting when the tree depth is excessive or when the halting condition is inadequately defined.

To overcome this problem, ensemble methods such as random forest or gradient boosting can be used to combine multiple decision trees and improve the performance of the model. The max_depth parameter is a hyperparameter that controls the maximum depth of a decision tree. The depth of a decision tree refers to the length of the longest path from the root node to a leaf node. Setting the max_depth parameter to a high value in a decision tree model can result in overfitting, where the model becomes excessively tailored to the training data. This can lead to the inclusion of noise and outliers that are unique to the training data but do not generalize well to new and unseen data[205]. On the other hand, if the max_depth is set too low, the decision tree may be too simple and perhaps leading to an inadequate representation of crucial patterns within the dataset. Hence, it is very important to be extremely careful while determining the max_depth parameter, to strike a balance between underfitting and overfitting. Overall, the max_depth hyperparameter controls the complexity of a decision tree and can be used to balance bias and variance in the model. The training and testing of these algorithms has been done in various steps. The methods are explained below:

1. **Out-of-Bag (OOB) Error Estimation**: An interesting aspect of Random Forest is its built-in method for testing called OOB error estimation. Since each tree is trained on a bootstrap sample, only about two-thirds of the training data is used for any given tree. The remaining one-third, which is not used during training, can serve as a test set. These left-out data points are called "out-of-bag" samples. By running these OOB samples through each tree and aggregating predictions, we can get an estimate of the test error without a separate validation set.

2. **Validation/Test Set**: Like any other machine learning model, you can split your dataset into training and test (or validation) sets. Once the Random Forest model has been trained using the training set, it is possible to assess its performance by evaluating its predictions on the test set, which consists of data that the model has not been exposed to during the training phase[133]. Common metrics for evaluation in classification tasks include accuracy, precision, recall, F1-score, ROC-AUC, etc.

3. **Feature Importance**: After training, Random Forest can provide a measure of feature

importance. This indicates which features are the most influential in making accurate predictions[164]. This can prove to be particularly advantageous for comprehending the model and for feature selection in datasets with a high number of dimensions.

4. **Interpretation**: While individual decision trees are interpretable, a Random Forest, being an ensemble of many trees, is harder to interpret directly[172]. However, tools and methods like SHAP (Shapley Additive explanation) and partial dependence plots can help provide insights into the model's decision-making process. Figure 3.8 shows the algorithm where the conditional sets are performed on piece of data received from the sensors.

For energy savings, the sensor data has been prioritized according to importance and integrity in relation to health of elderly people[199]. In a consecutive way, the iterations are performed to detect and execute the alarm of necessary information to be passed to a remote location. The information, as soon as received, thus enables the doctor/receiver to take immediate action as per the demand of the situation[197].

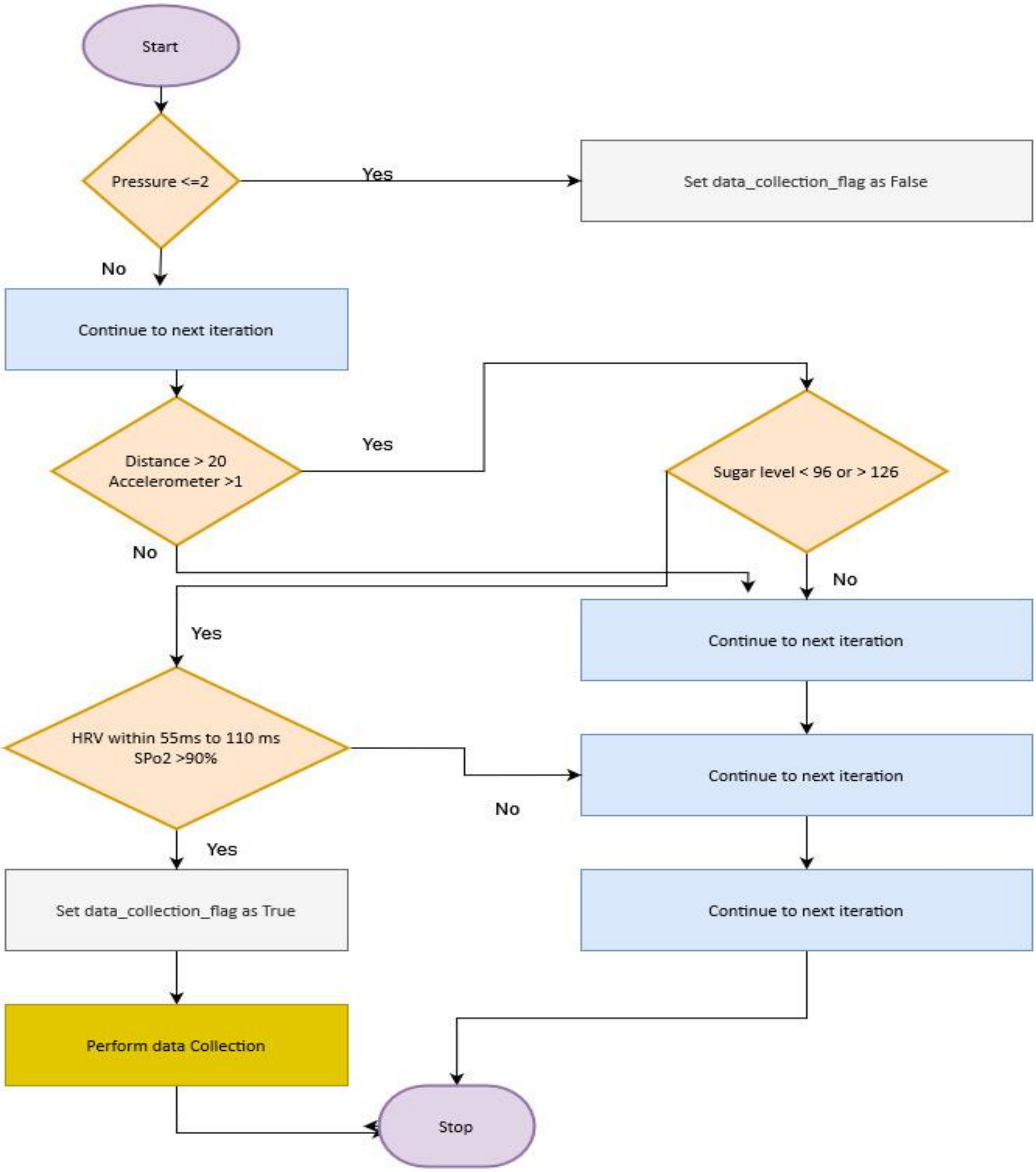

Figure 3.8: Algorithm for execution of result nodes as per conditions sets

3.4 Energy Efficiency Measurements

In this study, the LoRA based wireless module is used for data transmission and the machine learning algorithms are used for processing of data on edge scale for smart decision taking capability. Here, energy management and its scaling are observed as key factors for estimation and for job scheduling. Edge computing devices often operate in remote or decentralized locations[206]. Efficient energy use can significantly reduce operational costs, especially in environments where power sources are limited or expensive. It has been observed that the reduction in energy usage is closely correlated with a reduction in carbon footprint and subsequent environmental impact. This has significant importance, especially in the escalating worldwide efforts to address climate change. Energy-efficient devices typically generates less heat and may have longer lifespans[115]. This can improve the reliability and performance of edge computing devices, which is essential for critical applications like healthcare monitoring or autonomous vehicles.

As edge computing network grows, the cumulative energy demand can become substantial. Energy efficiency allows for more sustainable scaling of these networks. In many regions, there are increasing regulatory pressures to adopt energy-efficient technologies. Efficient edge computing solutions can help organizations comply with these regulations. Battery life and its maintenance are the key aims of edge computing applications. Energy efficiency metrics can help in better resource allocation and management. Understanding the energy profile of edge computing devices allows for optimizing workloads and network configurations for minimal energy use[207]. Measuring energy efficiency in edge computing can involve various factors, including the amount of computational work done, the energy consumed, and possibly other considerations like network latency or data throughput. Figure 3.9 shows the algorithmic process to measure the energy efficiency in relation to the modules in this study.

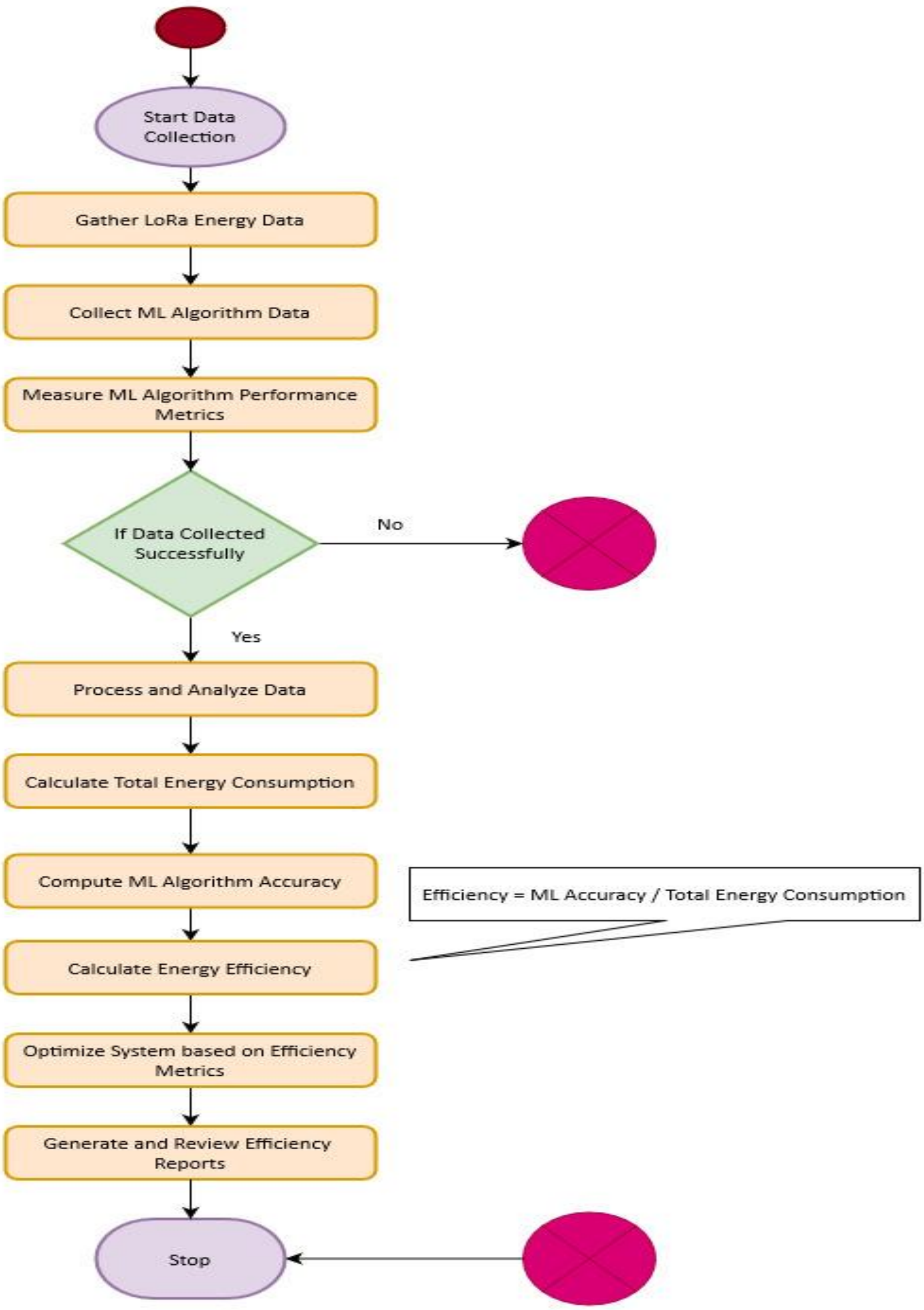

Figure 3.9: Algorithm for energy efficiency measurement in edge computing modules

Algorithm 1 Energy Efficiency Calculation for LoRa and ML Operations

```
Input: LoRaEnergyData: Array of energy consumption data for LoRa operations
Input: MLEnergyData: Array of energy consumption data for ML operations (RF & DT)
Input: MLAccuracyData: Dictionary with accuracy data for RF and DT
Input: TimeInterval: Time period for measurement
TotalEnergy ← 0
TotalAccuracy ← 0                    ▷ Initialize total energy and accuracy
for each energy in LoRaEnergyData do
    TotalEnergy ← TotalEnergy + energy
end for
for each energy in MLEnergyData do
    TotalEnergy ← TotalEnergy + energy
end for              ▷ Calculate total energy consumption for LoRa and ML
TotalOperations ← Length(MLAccuracyData["RF"]) + Length(MLAccuracyData["DT"])
TotalAccuracy ← (Sum(MLAccuracyData["RF"]) + Sum(MLAccuracyData["DT"]))/TotalOperations   ▷ Calculate weighted average accuracy for ML algorithms
if TotalEnergy > 0 then
    EnergyEfficiencyScore ← TotalAccuracy/TotalEnergy
else
    EnergyEfficiencyScore ← 0                 ▷ Handle division by zero
end if                                    ▷ Calculate Energy Efficiency
return EnergyEfficiencyScore          ▷ Return the Energy Efficiency Score
```

Based on the above-mentioned algorithm, the following equations are used to calculate the energy efficiency of the system.

The overall efficiency (E_{eff}) of edge computing application is given in equation 3.1:

$$E_{eff} = \frac{C_{work}}{E_{consumed}} \text{---(3.1)}$$

The C_{work} is the computational work accomplished in the edge module. This can be measured in terms of operations per second, tasks completed, or any relevant unit of computational work whereas $E_{consumed}$ is the energy consumed to perform job scheduling and throughput of data while transmission and processing, The energy consumption is basically measured in watt-hours (Wh) or joules (J). There are also some very important aspects such as network efficiency (N_{eff}) and data processing efficiency (D_{eff}) which needs to be integrated in the calculation of energy efficiency[72]. Equation 3.2 shows the relation:

$$E_{eff} = \frac{C_{work} \times N_{eff} \times D_{eff}}{E_{consumed}} \text{---(3.2)}$$

It is mandatory to compute the above parameters in relation to certain key processes whichare going in procession for edge computing-based application governance and provision of decision taking sorts.

Basically C_{work} is computational work in a computing system which is represented by various metrics, depending on the specific context and the nature of the tasks being performed (equation 3.3). It has been observed that there is a need to quantify computational work with the involvement of certain factors like the number of operations performed, the complexity ofthese operations and the time taken to complete them.

$$C_{work} = C \times T \times O \text{--------------------------------(3.3)}$$

Here *C* is the complexity of each operation including factors representing the average computational complexity of the operations. *O* represents the number of operations performed in the form of a number of the computational instructions executed. Here, *T* is the time taken to complete these operations[208].

The computation work can also be defined in terms of data processing scenario, it might involve the amount of data processed per unit time. Equation 3.4 shows the extended format of C_{work}:

$$C_{work} = \frac{Amount\ of\ Data\ Processed}{Processing\ Time} \text{---------------------------------------(3.4)}$$

During the calculation of computation power, it has been observed that the amount of data and processing time in relation to various complex situations and scenarios varied from a very minute to an exceeding level[209]. As per edge computing model complexity, such as size of training data, number of iteration cycles, execution of nodes etc. are also some associated parameters for measuring the energy efficiency of edge models as compared to other models. Similarly, the network efficiency (N_{eff}) has been measured by observing various contributing aspects such as throughput, bandwidth utilization, error rate, and latency[210]. Equation 3.5 shows the relation of network efficiency and other parameters:

$$N_{eff} = O \times L \times Th \text{------------------------------(3.5)}$$

Here, *Th* represents the effective throughput, which is the amount of useful data transmitted over the network in each period such as megabits per second. O is the overhead, which includes all additional data like headers, acknowledgments, and retransmissions due to errors that are necessary for transmission but are not part of the actual data. L is the latency, which is the delay in data transmission which is measured in milliseconds[211]. These aspects are playing a very important role in balancing the useful data transmission against the total data sent (including overhead) and the time delays experienced. Data processing efficiency (D_{eff}) isused to measure the effectiveness of the system to process the data relative to the resources consumed like time and computing power. A common way to conceptualize this is by considering the volume of data processed in each timeframe and the computational resources used for data realization[73]. Equation 3.6 represents the relation of network efficiency with various processing parameters:

$$D_{eff} = \frac{V_{processed}}{R_{consumed} \times T} \text{------------------------------------}(3.6)$$

Where $V_{processed}$ represents the volume of data processed as per the capacity of edge network and $R_{consumed}$ is the computational resources consumed depending on the CPU cycles, memory usage, or energy consumption such as CPU hours. T is the time taken to process the data in terms of seconds or hours. In local computing modules, such as edge computing devices and servers, the energy consumption ($E_{Edge\ N/W}$) is measured in the form of communication between all the devices. Equation (3.7) helps to sums up the energy consumed by each device or server for both computing and communication tasks.

$$E_{Edge\ N/W} = \sum_{i=1}^{n}(P_{compute,i} \times T_{compute,i}) + \sum_{j=1}^{m}(P_{comm,j} \times T_{comm,j}) \text{---------------}(3.7)$$

Here $P_{compute,i}$ is the power consumption of the i^{th} edge computing device/server during computation. $T_{compute,i}$ is the time duration for which the i^{th} device/server is involved in computation. $P_{comm,j}$ is the power consumption of the j^{th} device/server during communication activities. $T_{comm,j}$ is the time duration for which the j^{th} device/server is involved in communication activities. N is the total number of devices/servers involved in computation and m is the total number of devices/servers involved in communication. In relation to edge computing, equation (7) helps to expand the relations between the LoRA and machine learning algorithms for better estimation and optimization[212]. Equations 3.8, 3.9 and 3.10 shows the overall energy consumption (E_{Total}) relations in edge devices.

$$E_{Total} = E_{ML} + E_{LoRA} \text{ --(3.8)}$$

$$E_{ML} = P_{ML} \times T_{ML} \text{ --(3.9)}$$

$$E_{LoRA} = P_{LoRA} \times T_{LoRA} \text{ --(3.10)}$$

Here P_{ML} and P_{LoRA} represent the average power consumption of the machine learning computation steps and LoRa communication[72]. Whereas T_{ML} and T_{LoRA} show the time duration over which the computation and communication happened in the edge devices. The accumulative power consumption of Edge network is given in equation 3.11.

$$E_{Total} = (P_{ML} \times T_{ML}) + (P_{LoRA} \times T_{LoRA}) \text{------------------------------(3.11)}$$

Overall, various important variables such as throughput, network efficiency, search algorithm iteration time and latency rate are considered for the calculation of energy efficiency. In edge computing modules, efficiency represents the optimal use of energy resources for device sustainability and long-lasting performance in relation to applications[213]. It has been observed that the energy consumption of modules somehow depends on the sample of data to be processed, the history of decisions and the communication of the information to the end user without excessive latency.

3.5 Conclusion

The edge computing modules are very well equipped with such algorithm(s) which play(s) a pivotal role in managing energy usage. In these modules, the energy is consumed as per load demand optimization. The size of sample data and range of modules are the main load parameters in edge devices which are optimized as per the current and past scenarios of end user. Machine learning algorithms are very well versed to optimize the solution with a smaller number of iterative, which results in energy savings, and LoRA based low power modules are helpful in establishment of discrete communication over a long range with accuracy and precision. This is ultimately the aim of edge computing devices to adopt methods for improvement in energy efficiency in the application specific area. Further, the results are discussed in chapter 4. In results section, different situations were formulated and tested on the workbench. The results are presented to show the prediction accuracy, and certain cases were evaluated using classification methods. The correlation of actual and predicted variables was compared to scale out the achievement of adopted methods.

Chapter 4

Results and Discussion

4.1 Introduction

In edge computing devices, there are various modules which are responsible for efficient performance of system with strong capability of decisions and accurate execution of nodes in relation to outer environment data aspects. Machine learning tools are one of the key sections of modern decision taking machines for performance and integrity in outer environment with the help of sensors and actuators. The training, validation and testing is the key procedure of machine learning algorithms for accuracy in decisions in terms of activation or deactivation of functions, enabling or disabling the processes etc. In machine learning, the algorithmic approach is somewhere depending on the training data and its significant correlation with class of decision. The decision of a machine depends on various factors which are also called as features.

It has been observed in studies that the nature and trend of feature is very important. It is mandatory for a researcher to explore the features in terms of elements, dimensions, and trends so that the better control strategy many be designed and executed. In research studies, the type of features and number of features plays a pivotal role. As per features of data, advanced level of control strategies can be planned.

There are various fields such as health monitoring where the feature selection may improve the performance of AI model construction with other parameters such as interpretability, curse of dimensionality, quality of system, aids in future engineering. On the same verge, in this study, to design an edge computing powered energy efficient stick for patients under monitoring, six features were observed to be relevant and suitable for decision taking capability as desired. The features such as Heart Rate Variability, Sugar level, Oxygen saturation level (SpO2), Distance, Blood Pressure level and Axis based accelerometer decision were selected and combination of more than 40,000 situations were collected from real world for preparation of decision matric and training of machine learning algorithms.
It is very important to understand the features for further understanding of decision taking capability of device in real world situation. It has been observed that the trends in features

leads to understand various situations and decisions by the machine so that the best interpolation/extrapolation has been formulated and then the comparative analysis will perform for necessary decisions by the machine. It is also very significant to understand the range of data in terms of its normal and abnormal range so that the machine can compute the right decision for accurate activation of output nodes. Further the features are explained to understand the class wise distribution of data for its purity and genuinely to train the model.

4.2 Feature selection and data distribution

In this study, the data of primarily six features such as Heart Rate Variability, Sugar level, Oxygen saturation level (SpO_2), Distance, Blood Pressure level and Axis based accelerometer were collected from the patients to train the model. More than 40,000 combinational readings were measured for model training with robust scenario. The detail of each feature is given below to understand the statistical and temporal behaviors.

4.2.1 Heart Rate Variability

Heart Rate Variability (HRV) is basically a measure of the variation in time between each heartbeat. This variation is controlled by the autonomic nervous system (ANS) which is directly influenced by certain activities such as exercise, hormonal reactions, and stress levels[200]. HRV is an indicator of the balance between the sympathetic and parasympathetic branches of the ANS. Generally, the unit of HRV is number of Heart Beats Per Minute (BPM). As per studies, the HRV is categorized into two classes such as Normal and Abnormal range.

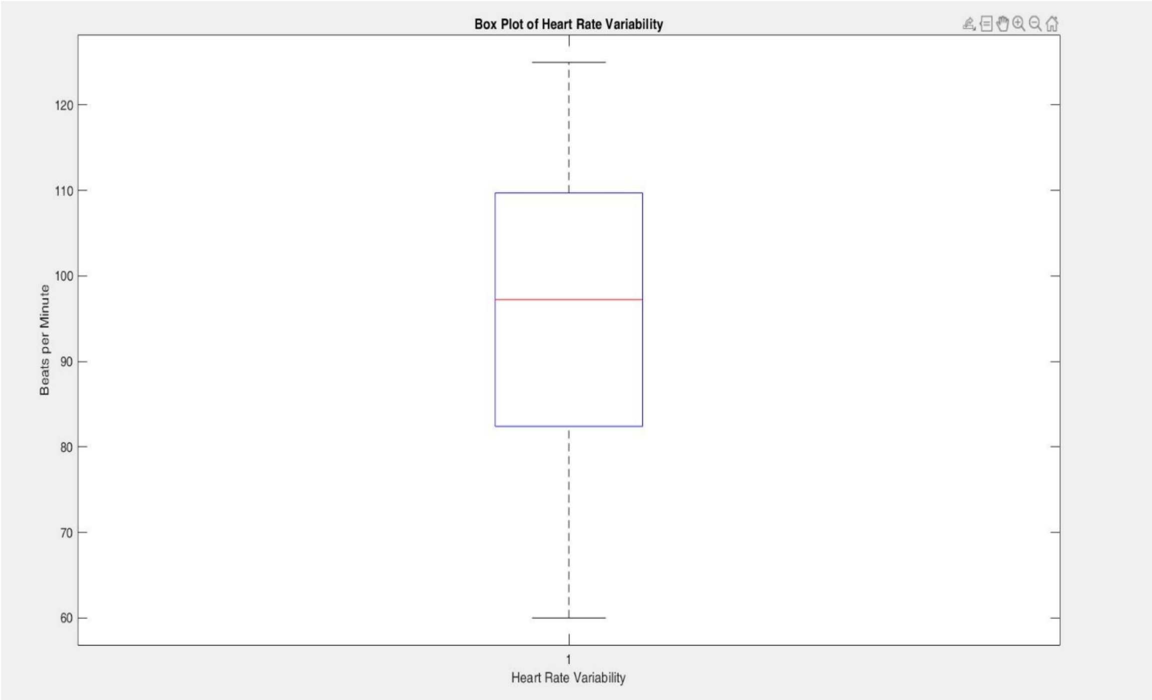

Figure 4.1: Box plot distribution of Heart Rate Variability Feature

For elder persons or patients, the normal range is varied from 60 to 90 BPM and above 90 BPM leads to abnormal condition. In this study, more than 21000 readings represented the normal HRV class under certain circumstances and more than 19000 readings presented abnormal HRV of persons under observation. The box plot of HRV data is shown in Figure 4.1. From figure, it has been observed that the box spans from the 25th percentile (Q1) to the 75th percentile (Q3) having 25th percentile (Q1) is 82.407, Median (Q2) is 97.238, the 75th Percentile (Q3) is 10 9.7025 with Interquartile Range (IQR) varied from 27.29 to 28.70. The overall data ranged from 60 BPM to 131 BPM having uniformity in both normal and abnormal class. There was no outlier data observed outside the range of the whiskers. The Histogram plot of HRV is shown in Figure 4.2.

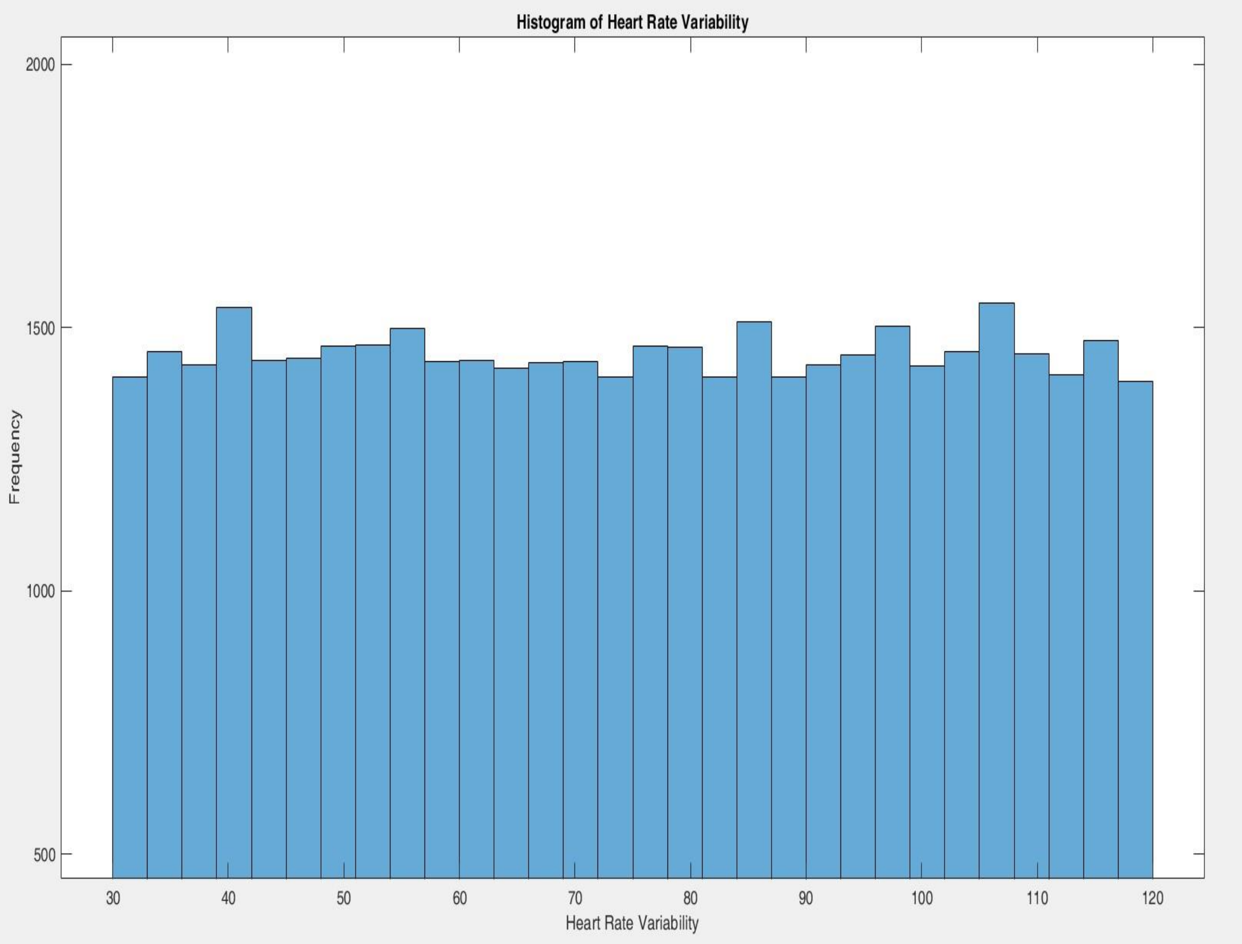

Figure 4.2: Histogram of Heart Rate Variability Feature

The histogram presents the frequency or distribution of collected data in certain ranges of HRV. It has been observed that from 60 to 90 BPM (Normal range), the frequency of data varied from 1300 to 1400 readings with cyclic increment of 10 BPM from 60 BPM. Whereas above 90 (Abnormal Range) the frequency of data varied from 1400 to 1550. The number of cyclic increments were bit more in normal range but the frequency of data is more in abnormal range. Hence the HRV data is uniformly distributed and have no value under outlier range.

4.2.2 Sugar Level

The sugar level or blood glucose level is one of the dynamic physiological parameters of human body. In patient monitoring and treatment process, this feature needs rigorous attention to prevent the emergency condition in relation to elder persons and patients[199]. The sugar level in the human body, typically measured in terms of blood glucose level. These levels are essential for diagnosing and managing conditions like diabetes. The sudden fall and rise in sugar level can cause some serious health issues in a short interval of time and can cause death. The sugar level is categorized into three main levels: low (less than 70 mg/dL), normal (70 to 100 mg/dL) and high (more than 100 mg/dL). In this study, the sugar level data have more than 40,000 combinational readings ranging from 10 to 180 mg/dL to cover the three main categories of sugar level. The Box plot of sugar level data is shown in figure 4.3.

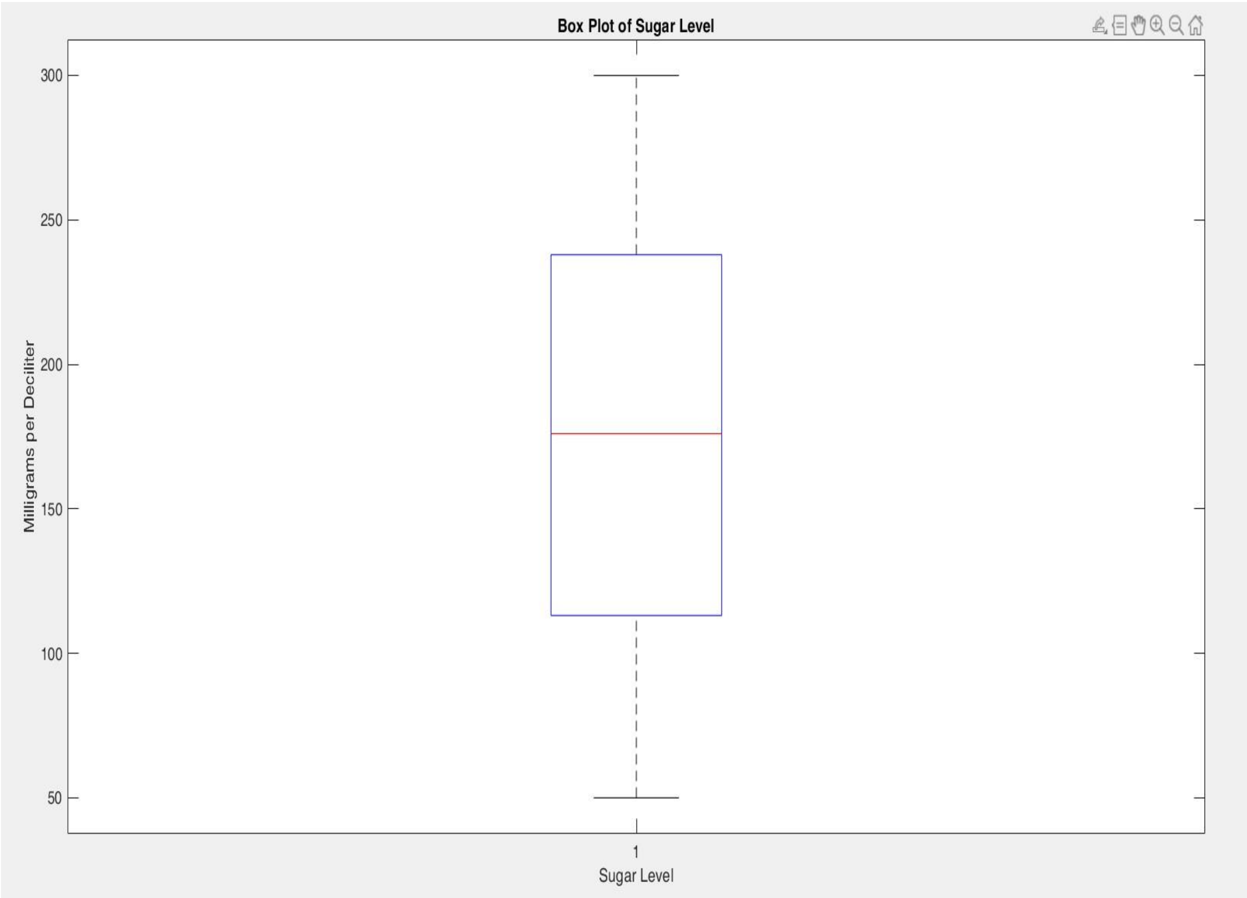

Figure 4.3: Box plot distribution of Sugar level feature

It has been observed from the box plot figure of sugar level that the data is varied from 50 to 300 mg/dL having 25th Percentile (Q1): 113.11, Median (Q2): 176.02, 75th Percentile (Q3): 237.92 and Interquartile Range (IQR): 124.81. The Histogram of sugar level data is presented in figure 4.4. The distribution of data in cyclic interval of range of sugar level shows the uniformity in categories of this feature.

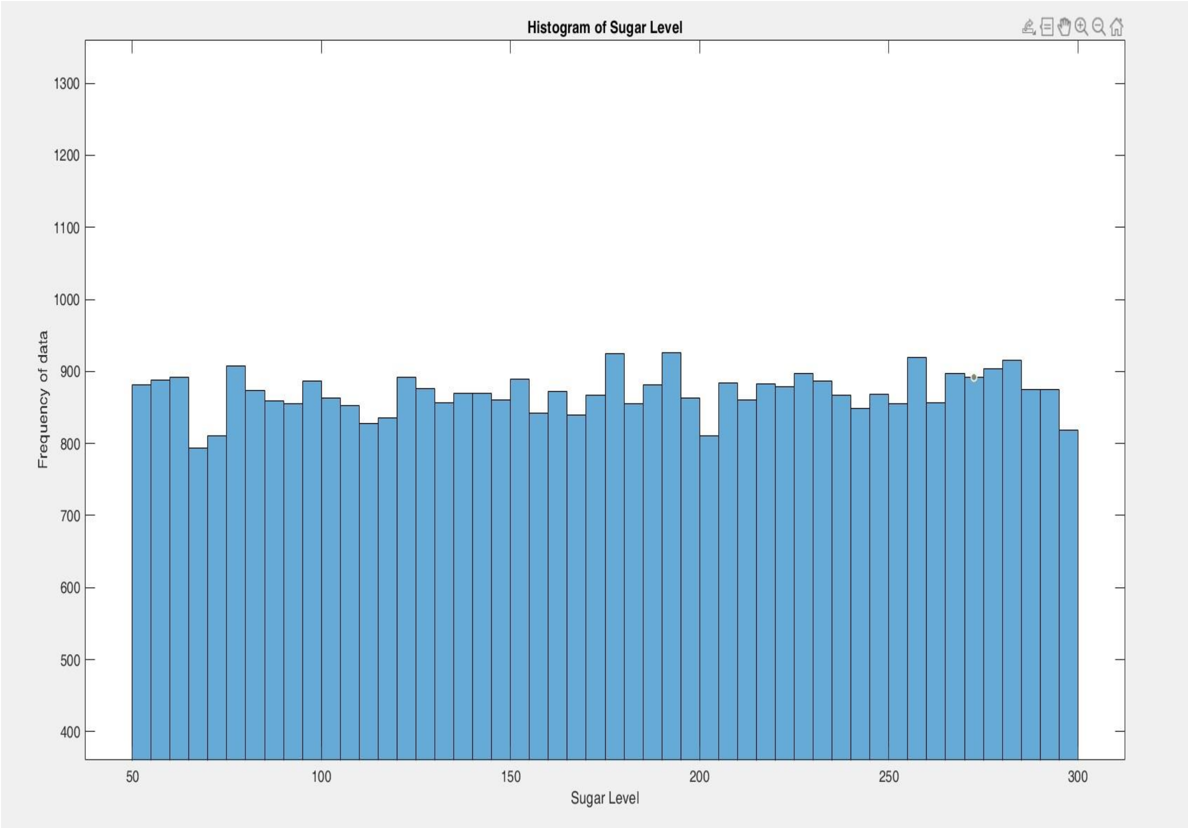

Figure 4.4: Histogram of sugar level feature

In the normal range of sugar level, more than 13000 reading lies. Similarly in low and high sugar level (Abnormal situation), the data is unilaterally distributed. By the histogram of sugar level, it may be observed that the low and high sugar level data has enough amount to satisfy the AI model building.

4.2.3 Oxygen blood saturation (SpO2) level

Oxygen saturation or SpO2, is a feature which is used to measure the amount of oxygen-carrying hemoglobin in the blood relative to the amount of hemoglobin not carrying oxygen. It is an important indicator of respiratory function and is often used in clinical settings to assess a person's oxygenation status[214]. This feature is also considered as one the integrated parameter of human physiology. This parameter is categorized into two classes normal and abnormal. The normal range of SpO2 varied from 85% to 100% whereas the level below 85% is considered as abnormal level. In this study, the box plot (figure 4.5) presents the quartile wise central tendency of data.

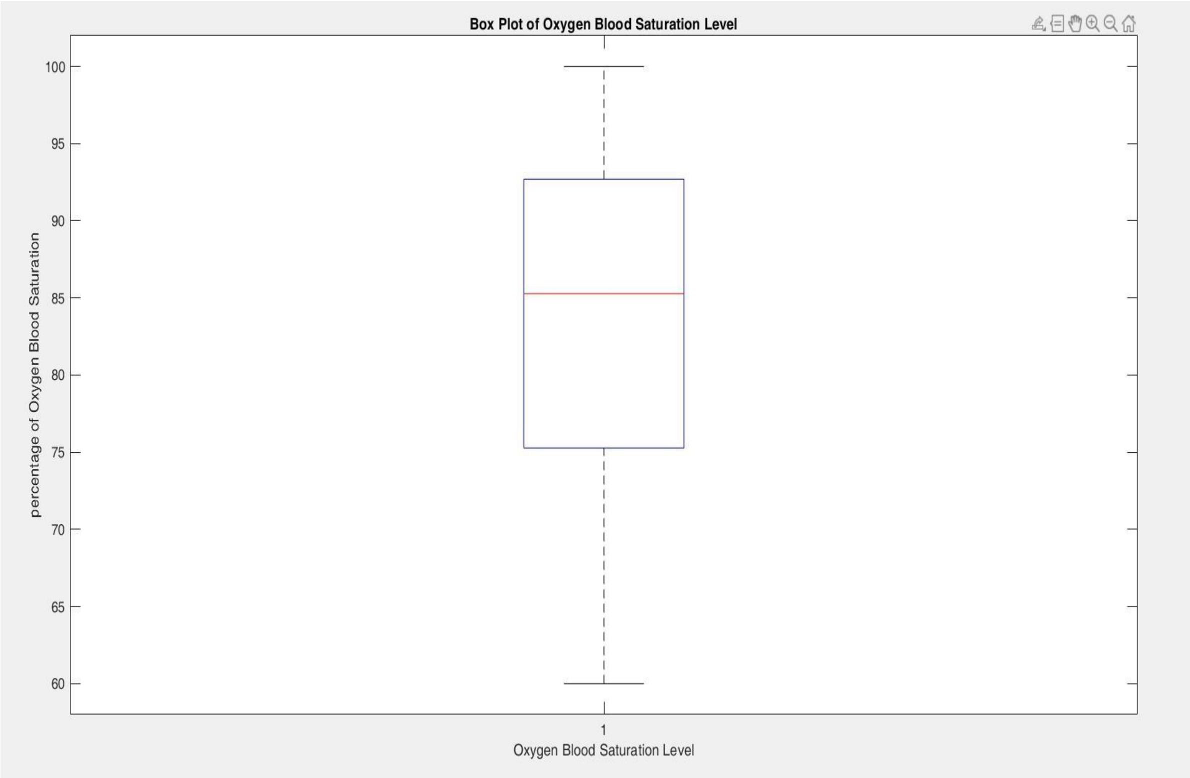

Figure 4.5: Box plot distribution of Oxygen blood saturation (SpO_2) level

The SpO2 data was observed to be measured from 60% to 100% in the observing candidates. The 25th Percentile (Q1) of observed data is 75.2775, Median (Q2) is 85.28, 75th Percentile (Q3): 92.6963 and Interquartile Range (IQR) is varied from 17.41 to 18.05. In the observed data there was no outlier data. The spread of data has median over 75% of normal range which is 15% lesser than the normal level. SpO2 parameter is a very stable parameter which shows less deviation from its consecutive values but under crucial stages, it may down due to respiration loss in the patients under monitoring. The Histogram of Oxygen blood saturation (SpO2) level is shown in figure 4.6. It has been observed from the figure that the data is ranged from 10000 to 23000 which covers almost both normal and abnormal range of data. The frequency distribution in abnormal category is 1000 with increment is cyclic increment up to normal range. Similarly, in the normal range, the frequency of data varied from 1500 to 2000 with increment in range by 5 values. By histogram, it has been observed that the data is uniformly distributed which helps the AI model to prepared the data base for prefect decision and preventing the confounding situation.

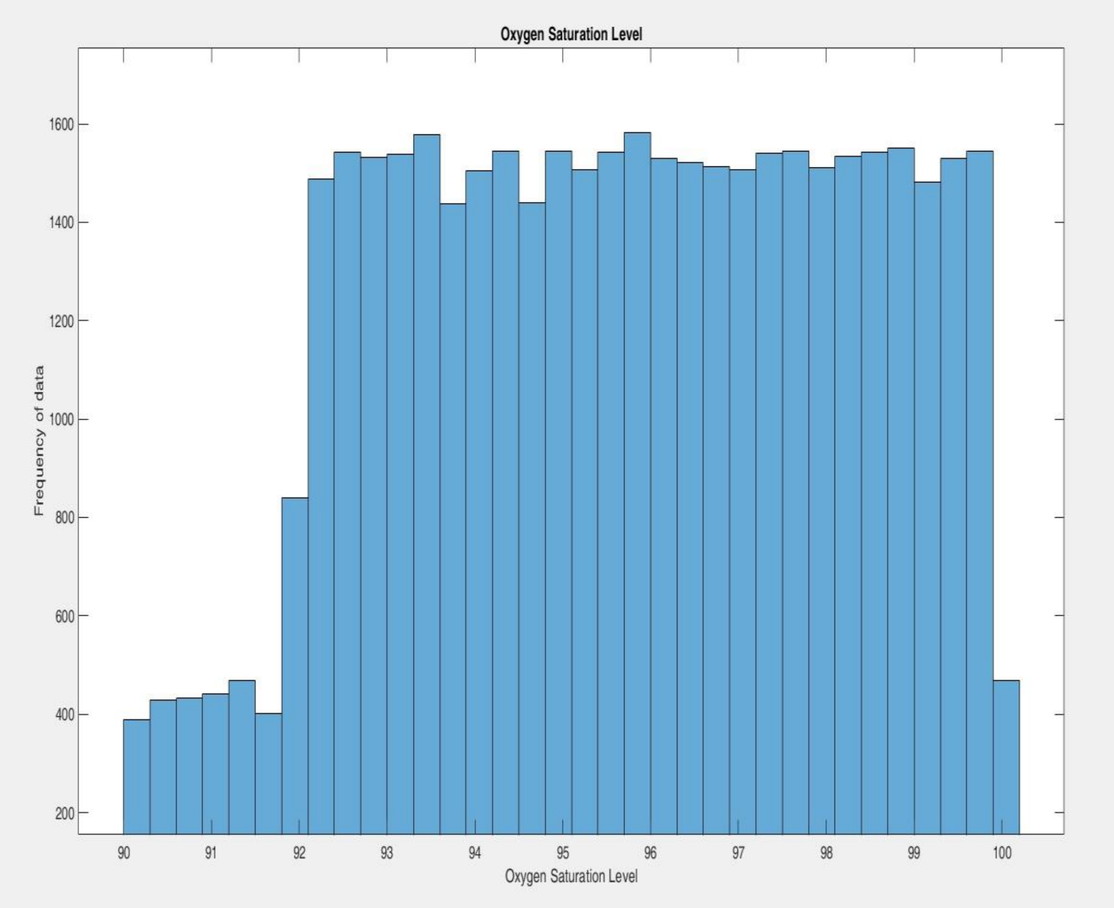

Figure 4.6: Histogram of Oxygen blood saturation (SpO_2) level

4.2.4 Blood Pressure (BP) level

In relation to human health, blood pressure is a crucial physiological signal that provides essential information about the health and functioning of the cardiovascular system. It is an indicator of the force exerted by circulating blood on the walls of blood vessels. The importance of monitoring blood pressure lies in its ability to indicate various health conditions and risks such as Hypertension, Heart attack etc. Like Heart rate variability and sugar level, the blood pressure signal is also an integrated signal of body that helps to indicate the chronic or normal situation of a patient or elder person[215]. As per reports, the blood pressure has three categories (Normal, Moderate and High) which are elaborated in term of systolic and diastolic blood pressure having unit's mm Hg (SP to DP mm Hg).

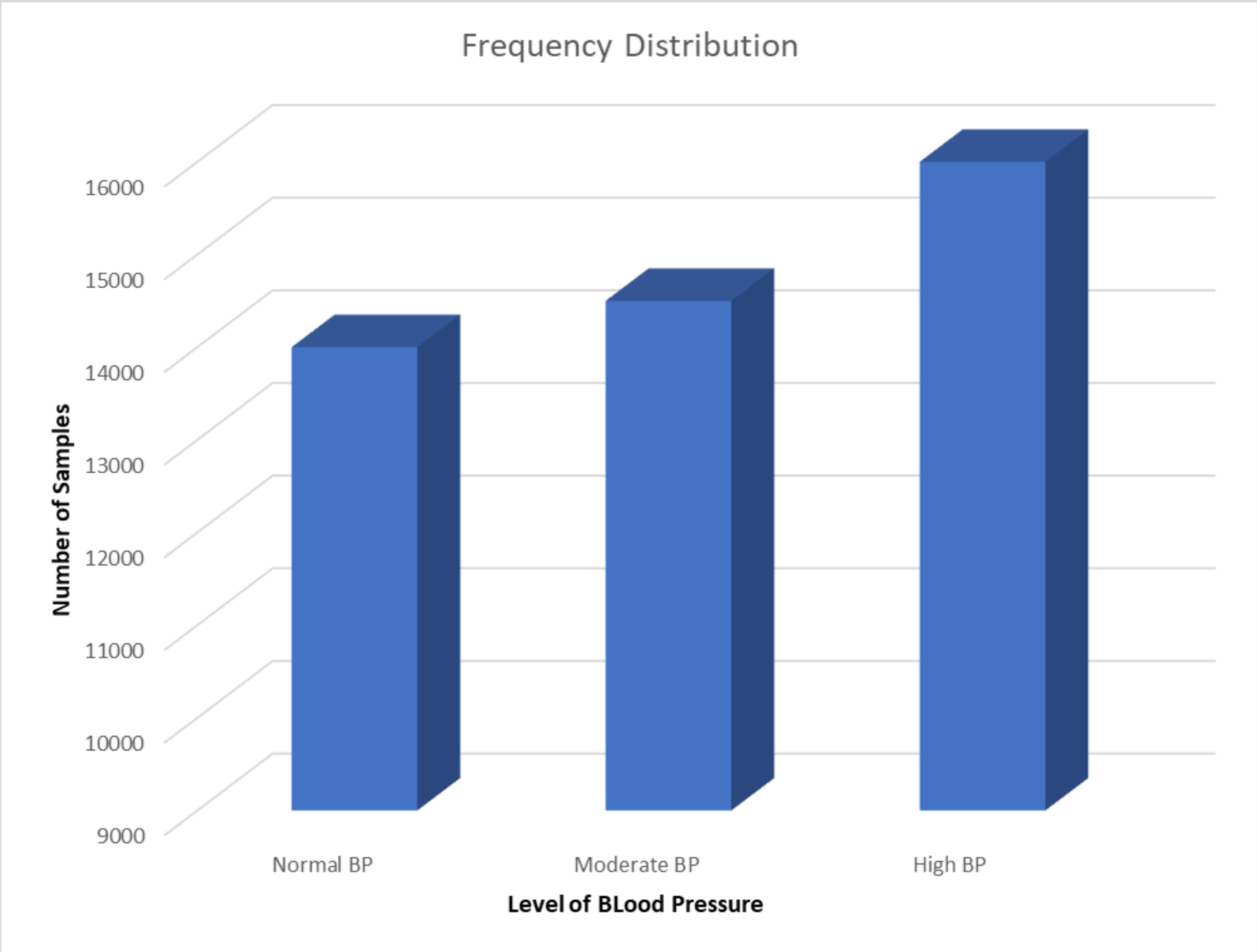

Figure 4.7: BP level wise number of samples

The range of normal blood pressure is >120 to >80 mm Hg and moderate blood pressure is 120-130 to 80-90 mm Hg. The high blood pressure is varying from above 140 - 90 mm Hg. In this study, like other features, the data of blood pressure is more than 40,000 and the levels of blood pressure was further re-categories in three numeric numbers as per the used sensor. As per sensor output, the normal blood pressure was indicated as 0, moderate blood pressure was indicated as 1 and high level of blood pressure was indicated as 2. So, the categories of blood pressures were recorded in the form of digits. The category wise data frequency is shown in figure 4.7. In normal range, there was more than 14000 entries and 15000 entries were recorded in the moderate level of BP. Under High blood pressure condition, there were also more than 16000 entries recorded for training and testing of AI models. As per numeric presentation of blood pressure, the data is somewhere normalized, which has various statistical benefits such as reduced computation time, improves latency, fast response and easy to interpret the information. Data is almost equally distributed in the three categories due to which the model overfitting will be reduced[215].

4.2.5 Accelerometer signal data and its category wise distribution

Accelerometer is an electronic sensor that measures acceleration forces in various forms. These forces can be static, like the constant pull of gravity or dynamic (caused by moving or vibration in the accelerometer).

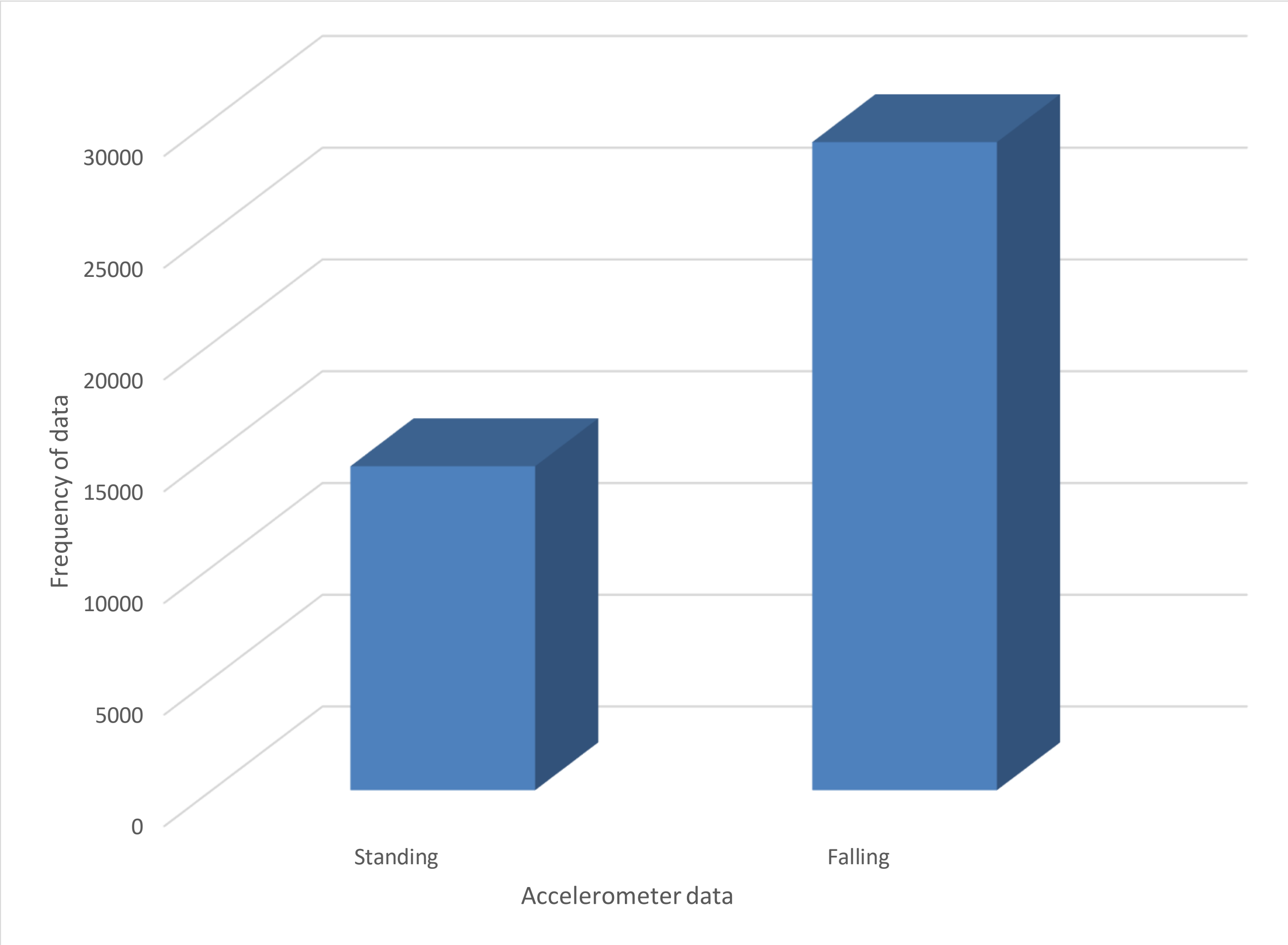

Figure 4.8: Category wise distribution of accelerometer sensor data

The accelerometer detects different patterns of acceleration corresponding to the person's movements. When the person is standing still, the accelerometer experiences gravitational force in a steady, unchanging manner, which is interpreted as a '0' reading. It indicates no significant motion or change in orientation. Conversely, a fall is a dynamic event characterized by a sudden change in acceleration and orientation. During a fall, the accelerometer detects a rapid shift in acceleration parameters, significantly different from the readings taken during standing[216]. This abrupt change triggers the sensor to record a '1', signaling a fall. This kind of detection is crucial for systems designed to provide immediate assistance or alert staff in cases where the person might be incapacitated or injured after a fall. The use of accelerometers for fall detection is particularly valuable in healthcare settings, especially for monitoring the elderly or individuals with conditions that increase their risk of falls. This electronic sensor is integrated into wearable technology or mobile devices, allow for continuous monitoring and potentially reducing the risk of severe injuries associated with falls. The binary system (0 for standing, 1 for falling) simplifies the data processing, enabling quick and efficient interpretation of the sensor's readings. In this study, an accelerometer data was also recorded from the same persons under observation and installed in the stick for real time measurement of data. The data distribution of standing and falling situation is shown in figure 4.8. In falling category (presented as 1) more than 28,000 readings and in standing category (presented as 0), more than 13000 readings were observed and recorded. In combination with other physiological features (which were recorded from the human subjects under monitoring), this feature was also measured and observed to be one of the key indicators of severe health situations of a person. The data of accelerometer is considered as addon feature for strengthen the decision results by the machine learning algorithms. This data is also free from any kind of outlier data which leads to true or positive computation abilities of an AI model.

4.2.6 Decision Indication and its correlation with measured features

Overall, for this study, total six features were recorded from the patients and more than 40,000 situations / readings were recorded to prepare the database. Based on the features, the medical practitioner also recorded the decision as outcome of the combinational features. The decision feature was categorized into three classes as per real scenario (no fall detected: 0, slip detected: 1 and fall: 2). Figure 4.9 is showing the data distribution of reported decisions with respect to other measured features such as Heart Rate Variability, Sugar level, Oxygen saturation level (SpO2), Distance, Blood Pressure level and Axis based accelerometer. It has been observed from the figure that in each category, more than 13,000 cases were reported. Based on measured features, the cases of abnormality were twice than the normal cases in which the person was still standing irrespective of minor changes in the physiological signals. But on the other side, the cases of slip and fall were more due to uncomfortableness in the patient’s health. There may be certain reasons such as abnormal rise or fall in HRV, blood pressure or sugar level or vice versa integrity in the features. As per reports, the physiological indicators are tied to each other in direct or indirect way. For instance, sudden rise in sugar level may cause over burden on blood pressure rate and heart rate variability of patients.

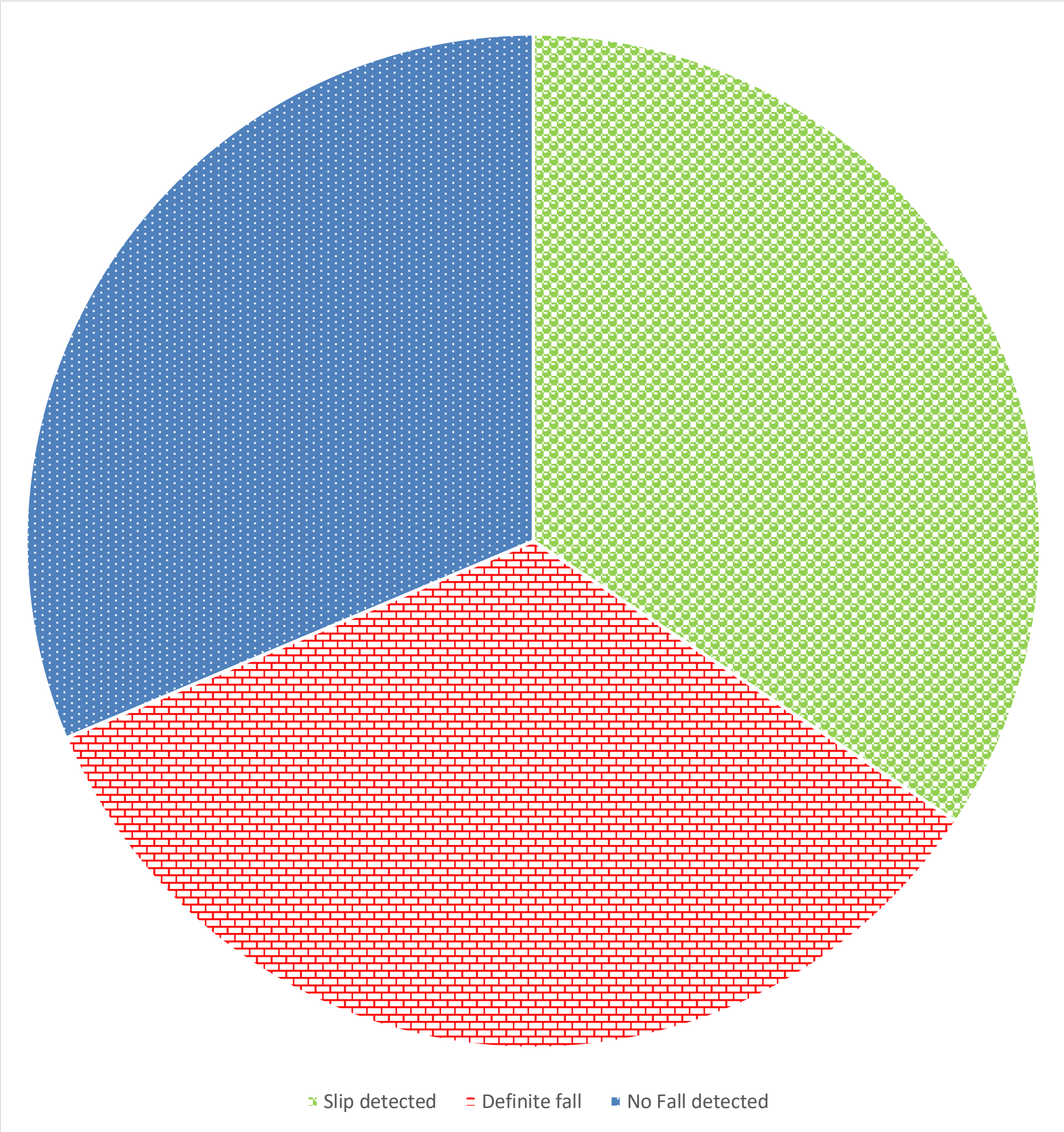

Figure 4.9: Data distribution of decision matric based on measured features

Here, it is very necessary to understand the impact of features in terms of rank and their correlation matrix with each other for estimations and preparation of control strategies. Figure 4.10 is showing the correlation heatmap between the decision and measured features.

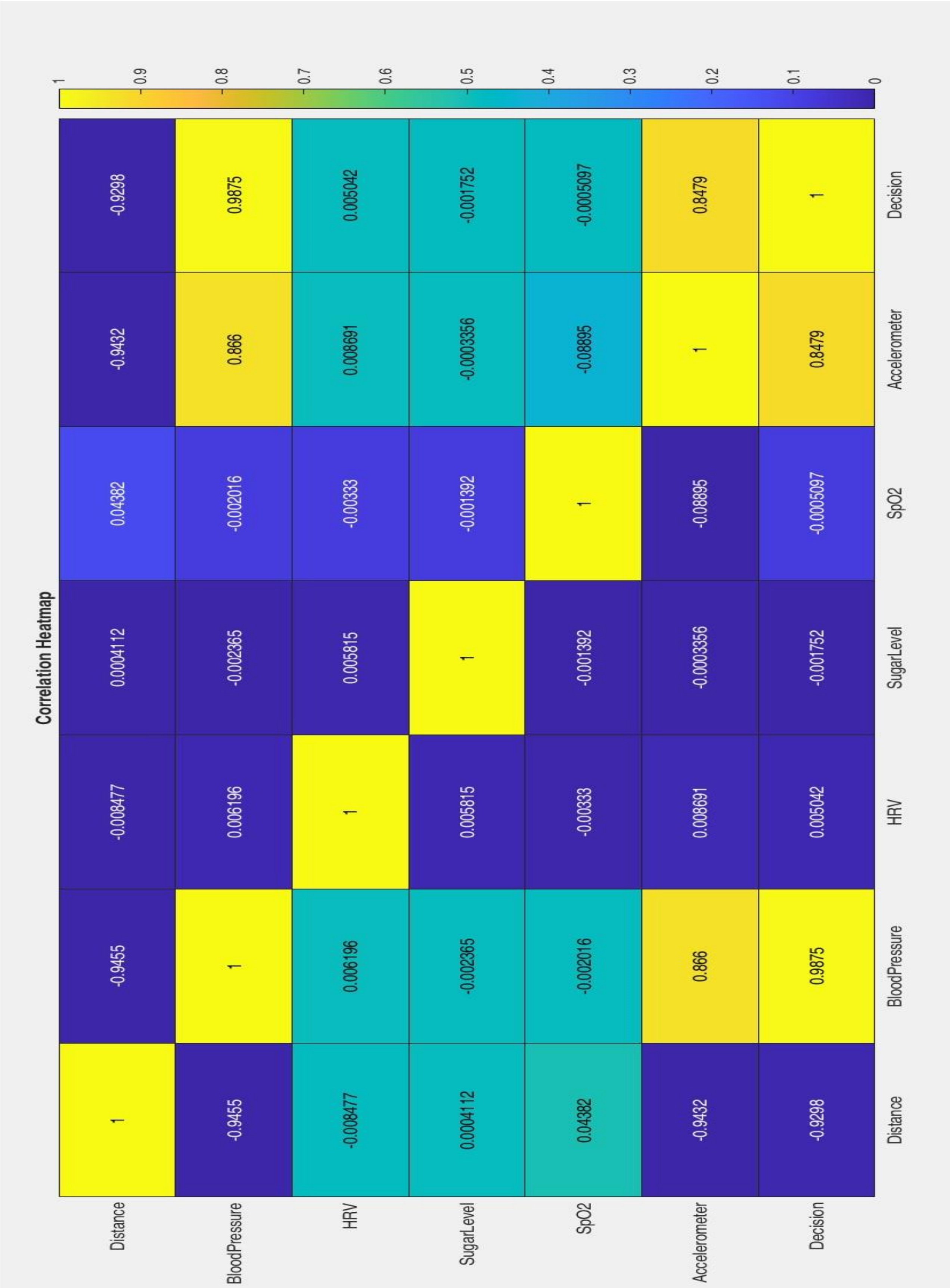

Figure 4.10: Correlation Heatmap matrix of measured features

It has been observed from the figure 4.10 that the blood pressure signal has a significant positive correlation with Heart rate variability feature (0.952) and accelerometer signal (0.8669) and the correlation between blood pressure data and decision is also very strong. It is confirmed from the correlation values that the blood pressure is contributing a pivotal role in the right decision observation. Heart rate variability has also positive and strong correlation with blood pressure parameter and accelerometer data which leads to a strong bond with decision parameter of respective samples. As per heat map, the elevated sugar level has positive correlation with raised heart rate variability in the human subjects. It has been observed that the sugar level of patients has direct impact on heart beat counts due to which the HRV and BP parameters varied. Similarly, Oxygen saturation level (SpO2) has very strong negative correlation with BP and Heart rate variability which means that an abnormal condition may be reflected in the SpO2 level of human subjects.

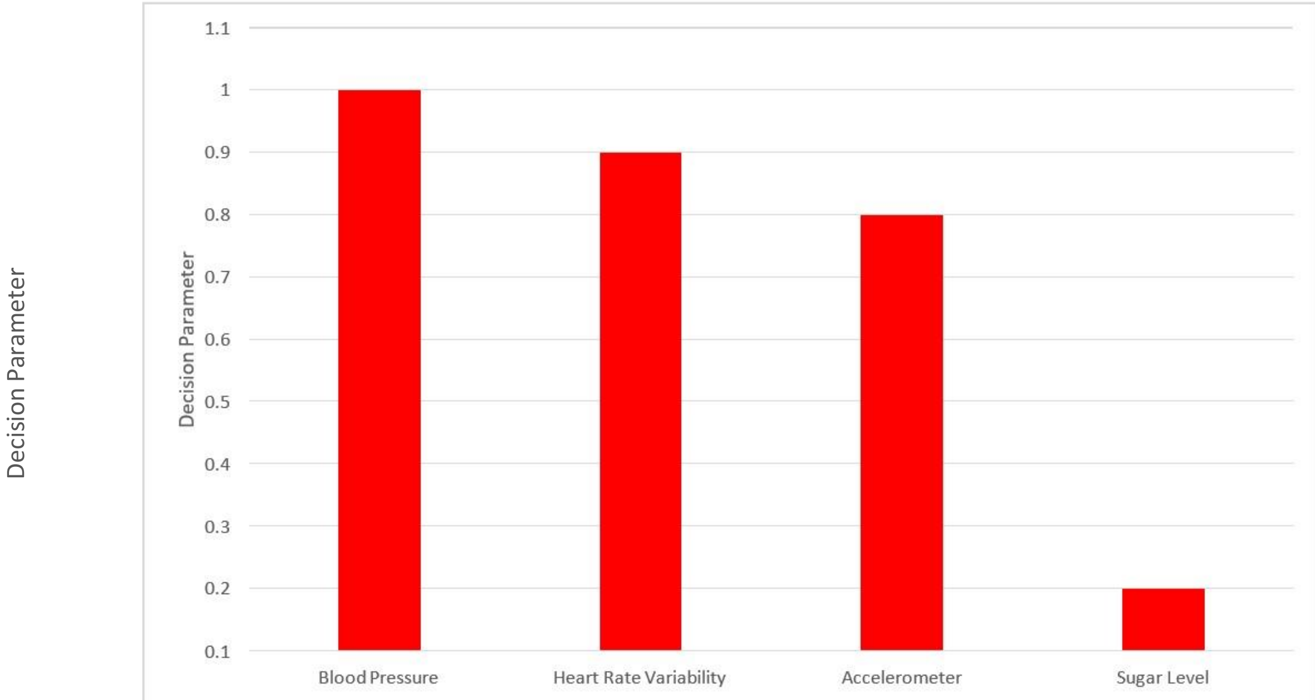

Figure 4.11: Rank wise relation of measured physiological features with decision parameter

From the matrix it has been observed that the measured accelerometer data has very positive correlation with physiological features such as BP and HRV and it is positively contributing in decision parameter. So, it is very clear from the heatmap correlation matrix that the measured physiological features have very strong correlation information of decision parameter and have integrated impact on each other for indication of normal and abnormal condition of an observant.

Further, figure 4.11 is showing the rank of features in relation to decision parameter of data base. It has been observed from the figure that the blood pressure is ranked first with very strong contribution in decision parameter. Consecutively, the Heart rate variability (Pulse rate) has positive correlation (0.9) with decision parameter. The HRV and blood pressure has very less difference in their rank. The accelerometer data has also ranked third with correlation of 0.8 with respect to decision parameter. In next to the order, the sugar level is ranked fourth having correlation with decision parameter. It has been observed from the figure 4.10 and 4.11 that the decision parameter has very strong correlation with measured features reported in the database. The correction matrix and rank of features helps to decide the priority of sensors to scan the condition of a patient and helps to save the energy of complete module. Figure 4.12 is showing the priority of sensors to monitor and activate the consecutive loop of sensors for saving the energy of module.

As per rank of features, the Pulse rate sensor is prioritized as sensor 1 which needs to be active and monitor the abnormality in the patient under observation keeping rest sensors under sleep mode. As per detection of abnormality in the pulse rate it will further activate the sensor 2 and sensor 3 for observation of concerned parameters such as bloop pressure of patient and then the accelerometer for detection of abnormality. Further, in sequence, the SpO2 and other sensors will get activated as per situation of a patient under observation. This activation sequence will also save energy and time consumption. Further this data will be shared with prediction modules for judgement. In this way, the sensor loop will get activated as per conditional priority of a patient due to which the health situation of patient will be monitored properly. Figure 4.12 is one of the best methodologies to adopt the sensors in the scanning loop for job Scheduling with the management of energy consumption.

Also, due to this correlation factors, it can be strongly hypothesized that the measured features have genuine relations to decision parameter and further the data can be used to train the machine learning modules for training, validation and testing and further prediction of decisions on real time measured feature values from deputed sensors. Based on the database, the control strategy is further discussed in relation to implementation of edge computing modules with the help of machine learning tools.

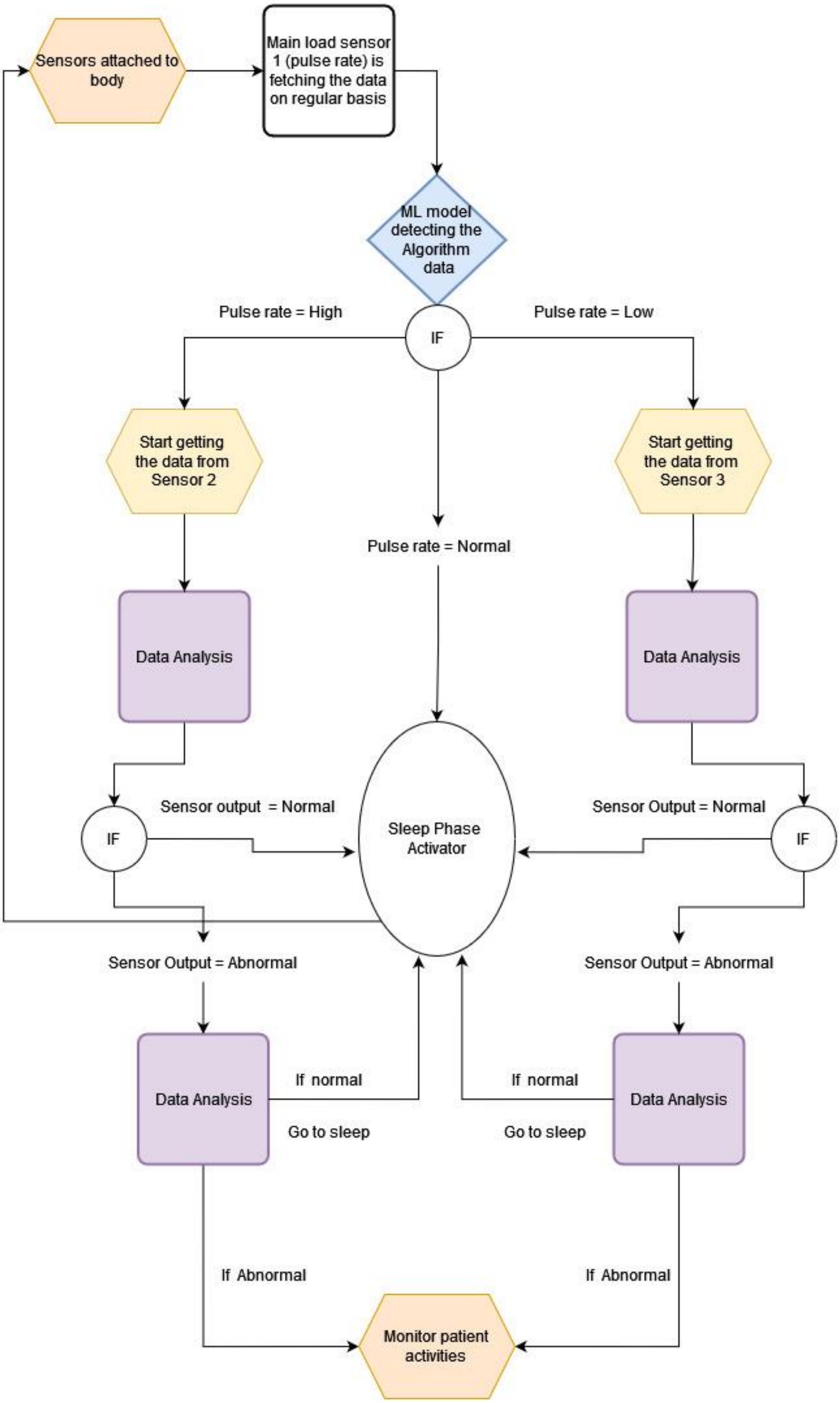

Figure 4.12: Patient monitoring strategy as per rank of input features

4.3 Training and testing of machine learning classification algorithms

Machine learning tools are sophisticated and robust algorithm that falls under the category of ensemble learning techniques. They are best working in supervised mode for prediction and outlining the decisions. It has been observed that these tools need the training data with specified attributes with its outcome parameters for training, validation, and testing, so that they can predict the outcome based on the history of cases. As per initial experimental trails, various algorithmic approaches were applied to train and test the data in relation to trade off based on time space complexity. As per observation, the random forest, decision tree and SVM classifiers were performed well due to their high level of accuracy and energy management during computation. In this study, these algorithms were combined and trained accordingly (Figure 3.1).

4.3.1 Training and testing of Decision Tree Algorithm

A Decision Tree Classifier is also a popular supervised machine learning algorithm used for classification tasks. It operates by splitting a dataset into smaller and more homogeneous subsets based on differentiating features, forming a tree-like model of decisions. The process starts at the root node and involves evaluating attributes to make binary or multi-way splits at each node. These splits form branches leading to further nodes or to leaf nodes, which represent the final classification outcomes. Decision trees are favored for their simplicity and interpretability, as they mimic human decision-making processes and can be visualized easily. However, they are prone to overfitting, especially in cases of complex trees or noisy data. To counter this, techniques like pruning (removing sections of the tree that provide little predictive power) or using ensemble methods such as Random Forests are often employed. Decision trees can handle both numerical and categorical data, making them versatile for various applications. Figure 4.13 is showing the algorithm of decision tree classification method. Table 4.1 is showing the values of performance evaluation parameter.

Table 4.1: Performance parameters of decision tree algorithm

S. No.	Parameter	Value
1	Number of Input parameter	6
2	Number of Output parameter	1
3	Sample number in each parameter	43000
4	Hold out data for testing	30%
5	Model name	fitctree
6	Best Depth	2
7	Precision	96%
8	Recall	96.2%
9	F1 Score	94.3%
10	AUC-ROC:	95.7%

The collected data was processed through decision tree algorithm for achievement of best performance parameters. The Fitctree model was selected with prediction of best depth level having highest accuracy. After training and testing it has been observed that the best depth level was 2 having 94.3% F1 score and the Recall was also 96.2%.

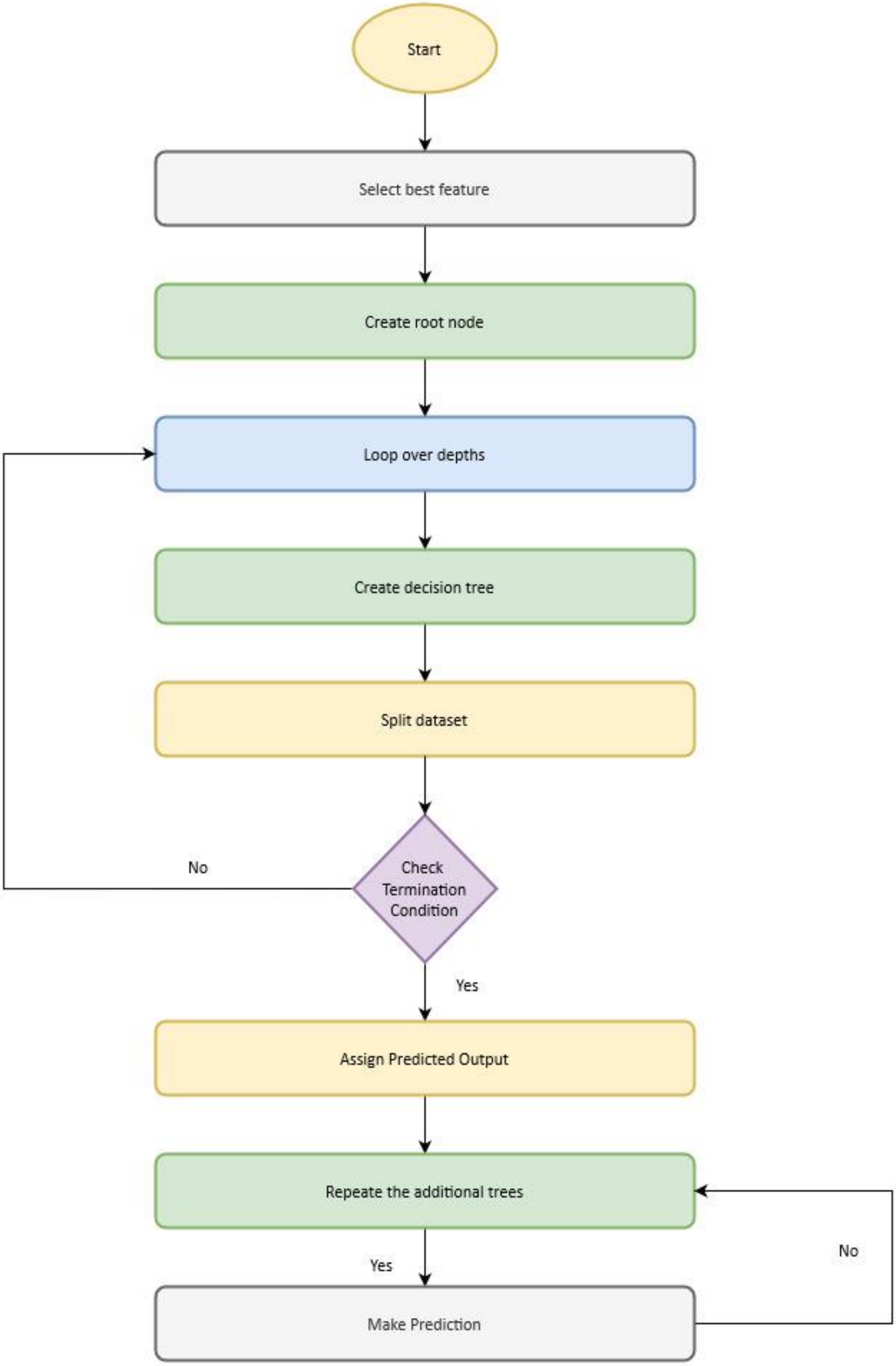

Figure 4.13: Decision tree classification algorithm for EoT based energy efficient decisions

The Table 4.2 is showing the achievement of accuracy in prediction of right decision by the decision tree machine learning classification algorithm.

Table 4.2: Prediction accuracy of Decision tree algorithm for real time sensor data

S. No.	Distance	BP	HRV	SL	SpO_2	Accelerometer	Predicted	Actual
1	28	1	123.24	45.89	96.45	1	Slip detected	Slip detected
2	45	0	82.61	45.6	98.05	0	No Fall detected	No Fall detected
3	28	1	120.57	200	84.6	1	Definite fall	Definite fall
4	20	2	78.87	23.44	83.2	0	Definite fall	Slip detected
5	80	1	23.24	175.8	91	1	Definite fall	Slip detected
6	63	0	13.24	60.46	91.3	0	Definite fall	Definite fall

Out of total six cases, in two cases, the decision tree has given wrong prediction. Overall, the accuracy of this classifier varied from 85% to 90%.

4.3.2 Training and testing of Support Vector Machine Algorithm

A Support Vector Machine (SVM) classifier is one of the supervised machine learning algorithms used for classification tasks. It operates by splitting a dataset into smaller and more homogeneous subsets based on differentiating features, forming a tree-like model of decisions. The process starts at the root node and involves evaluating attributes to make binary or multi-way splits at each node. These splits form branches leading to further nodes or to leaf nodes, which represent the final classification outcomes. Figure 4.14 is showing the algorithm of SVM classification method. Table 4.3 is showing the values of performance evaluation parameter.

Table 4.3: Performance parameters of Support Vector Machine algorithm

S. No.	Parameter	Value
1	Number of Input parameter	6
2	Number of Output parameter	1
3	Sample number in each parameter	43000
4	Hold out data for testing	30%
5	Model name	fitcsvm
6	Kernel Type	Linear
	Regularization Parameter (C)	95%
7	Precision	97.22%
8	Recall	96%
9	F1 Score	96.69%

The gathered data was processed using the Support Vector Machine (SVM) algorithm to achieve optimal performance parameters. The fitcsvm model was chosen, focusing on the appropriate kernel and regularization parameter (C) to ensure the highest accuracy. Upon training and testing, it was observed that with the fine-tuned parameters, the SVM model achieved a high level of precision. The F1 score reached an impressive 96.69%, and the model also demonstrated a Recall of 96%, indicating its robustness and effectiveness in classification tasks.

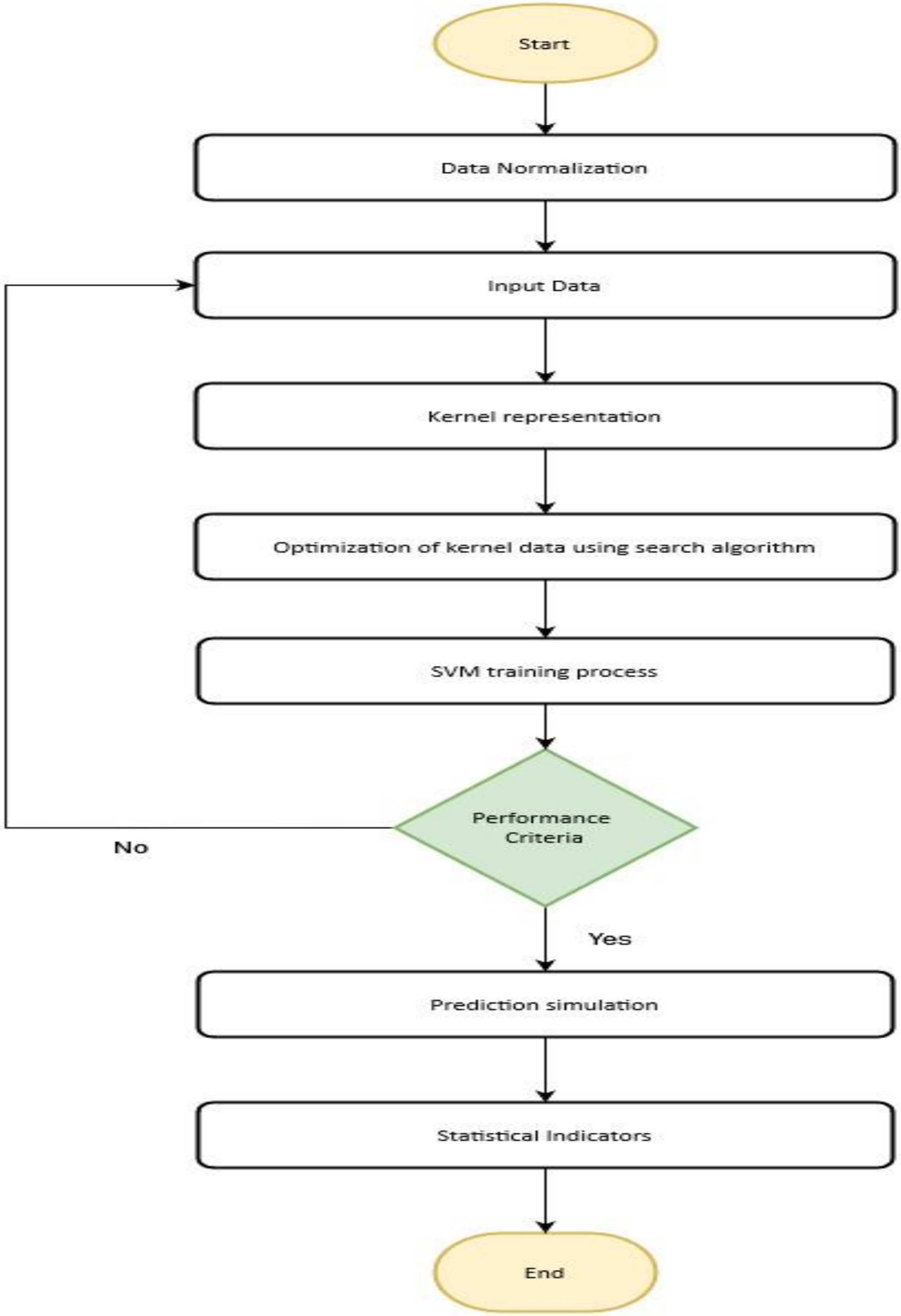

Figure 4.14: Support Vector Machine classification algorithm for EoT based energy efficient decisions

The Table 4.4 is showing the achievement of accuracy in prediction of right decision by the Support Vector Machine learning classification algorithm.

Table 4.4: Prediction accuracy of Support Vector Machine algorithm for real time sensor data

S. No.	Distance	BP	HRV	SL	SpO2	Accelerometer	Predicted	Actual
1	28	1	123.24	45.89	96.45	1	Slip detected	Slip detected
2	45	0	82.61	45.6	98.05	0	No Fall detected	Slip detected
3	28	1	120.57	200	84.6	1	Definite fall	Definite fall
4	20	2	78.87	23.44	83.2	0	Definite fall	Definite fall
5	80	1	23.24	175.8	91	1	Definite fall	Slip detected
6	63	0	13.24	60.46	91.3	0	Definite fall	Definite fall

Out of total six cases, in two cases, the Support Vector Machine has given wrong prediction. Overall, the accuracy of this classifier varied from 96% to 96.89%.

4.3.3 Training and testing of Random Forest Algorithm

Random Forest algorithm belongs to an ensemble machine learning algorithm that can builds multiple decision trees and merges them together to get a more accurate and stable prediction. It is particularly useful for classification and regression tasks and known for its effectiveness in handling large datasets with higher dimensionality, robustness against overfitting and its ability to improve accuracy by averaging the results of individual trees. Essentially, it constructs a 'forest' of decision trees during training and its output is determined by the collective output of these trees, which makes it highly effective and accurate for classification and regression tasks.

Table 4.5: Performance parameters of random forest algorithm

S. No.	Parameter	Value
1	Number of Input parameter	6
2	Number of Output parameter	1
3	Sample number in each parameter	43000
4	Hold out data for testing	30%
5	Model name	Tree Bagger
6	Number of trees	30 to 200
7	Best number of trees for best accuracy	50
8	Accuracy achieved	97.89%
9	Precision	96.35%
10	Recall	97%
11	F1 Score	97.4%
12	AUC-ROC	96.97%

The core principle of Random Forest involves constructing multiple decision trees during the training phase. Each tree is built from a randomly selected subset of the training data, a method known as bootstrap aggregating, or bagging. This randomness ensures that the trees are diverse, reducing the risk of overfitting and improving the model's generalization capabilities. In building each tree, Random Forest randomly selects a set of features at each decision point (node). This random selection of features contributes further to the diversity among the trees in the forest. Unlike some algorithms that prune trees to avoid over-complexity, trees in a Random Forest are typically grown to their full depth, allowing the model to capture complex patterns in the data. During prediction, the Random Forest algorithm takes an elaborative approach (each tree in the forest makes a prediction, and the final output is determined by combining these predictions). In classification tasks, this algorithm considers majority voting system, where the most common prediction among all trees is chosen as the final output. This ensemble approach significantly boosts the model's accuracy and reliability, as it mitigates the errors of individual trees. Figure 4.15 is showing the algorithmic approach to train the random forest model. The table 4.5 is showing the achievement of performance evaluation parameters.

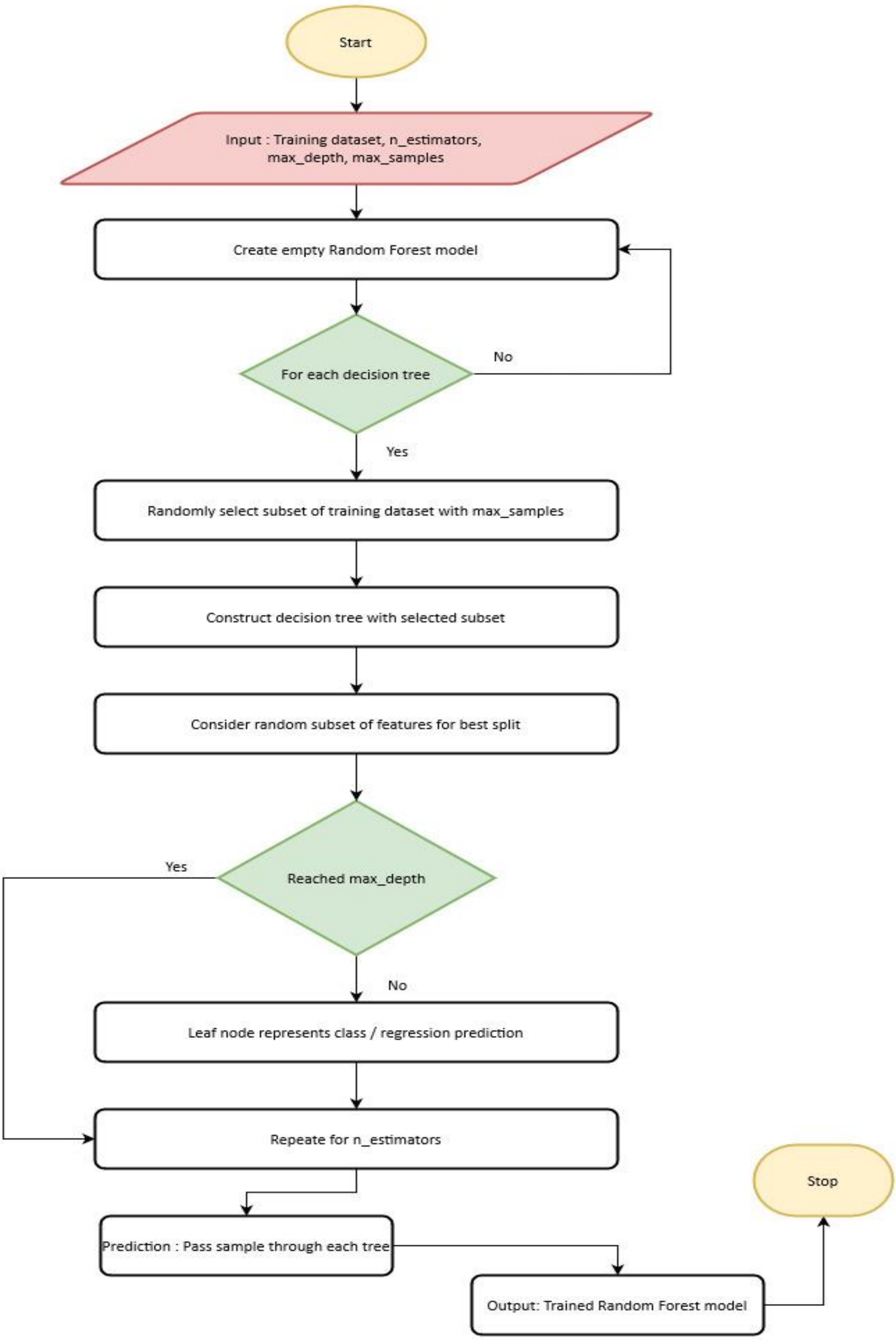

Figure 4.15: Random Forest classification algorithm for EoT based energy efficient decisions

As per collected data, the tree bagger function of random forest was selected with initial number of 30 trees and randomly increased to up to 200 for observation of best accuracy with prevention of overfitting of model. As per results, the 97% accuracy and relative F1 score was achieved at 50 trees. After training and testing, the real time sensor data was given to algorithm to test is prediction capability. The table 4.6 is showing the achievement of accuracy in prediction of right decision.

Table 4.6: Prediction accuracy of Random Forest algorithm for real time sensor data

S. No.	Distance	BP	HRV	Sugar Level	SpO2	Accelerometer	Predicted	Actual
1	28	1	123.24	45.89	96.45	1	Slip detected	Slip detected
2	45	0	82.61	45.6	98.05	0	No Fall detected	Slip detected
3	28	1	120.57	200	84.6	1	Definite fall	Definite fall
4	20	2	78.87	23.44	83.2	0	Definite fall	Definite fall
5	80	1	23.24	175.8	91	1	Definite fall	Definite fall
6	63	0	13.24	60.46	91.3	0	Definite fall	Definite fall

The accuracy of random forest algorithm was varied from 97% to 97.89% for real situation data. Which is observed to be better than decision tree algorithm and SVM. As per training and testing of random forest, decision tree and SVM, all the selected algorithms were deputed for generation of decision. The data was floated to algorithms at a time for training, validation and testing and then based on highest accuracy in association with time consumption during training and execution of result, the decision of that algorithm was executed (Figure 4.12). During the selection of machine leaning algorithm, some key aspects were taken into consideration such as processing or computation time to reduce the latency. The decision tree and random forest algorithms are deputed due to their less computation time than other algorithms and have multicategory decision characterization property than Support Vector

Machine.

4.4 Measurement of Energy consumption by the wireless models and their comparative analysis in relation to edge computing

In this study, the main purpose of wireless communication models was to establish a long range, energy efficient and latency free communication between the data collection module and local edge computing module for real time decision taking capability to save the life of a human subject. The data collection from sensor module was initiated as per level of conditional emergency flag (Figure 3.9). Here the WiFi and LoRA module was used for establishment of communication and compared for best performance. The transmission time, energy consumption etc. in Long Range communication system for a given amount of data depends on several factors. Here various parameters are observed for energy management in terms of edge computing[78].

i) Spreading Factor (SF): This parameter is one of the leading factors in relation to energy management. As per reports. higher spreading factors increase transmission time but also enhance range and robustness.

ii) Bandwidth: It has been observed that the narrower bandwidths result in longer transmission times, but can also increase the receiver's sensitivity.

iii) Coding Rate: As per sensor data, a higher coding rate adds more redundancy to the data for error correction, which increases transmission time.

Based on above mentioned factors, the WiFi and LoRA modules are compared. Table 4.7 is showing the comparison.

Table 4.7: Comparative analysis of WiFi and LoRA modules

Feature’s	**LoRa**	**WiFi**
Range	Long (up to 15-20 km)	Short (typically 100m indoors)
Data Rate	Low (0.3-27 kbps)	High (up to 1-7 Gbps)
Frequency Band	Sub-GHz (868/915 MHz)	2.4 GHz, 5 GHz
Power Consumption	Very low	Higher
Network Topology	Star, star-of-stars	Point-to-point, multipoint
Interference	Less prone due to Sub-GHz band	More prone due to crowded bands
Complexity/Cost	Low complexity, cost-effective	More complex, higher cos
Latency	Higher (due to low data rate)	Lower
Security	Basic, suitable for low-risk applications	Advanced (WPA3, etc.), suitable for high-risk applications

Based on the comparative analysis, the LoRA module was found to be best alternate due to its less power consumption. The technical specification of LoRA is given in table 4.8.

Table 4.8 LoRA Module parameters and transmission time observations

S. No.	**Feature**	**Value**
1	Spreading Factor (SF)	7
2	Bandwidth	125 kHz
3	Payload Size	10 bytes
4	Preamble length	8 symbols
5	Coding Rate	4/5
6	Explicit Header Mode	Enabled
7	Approximate Transmission Time	Less than 51 milliseconds

The figure 4.16 is showing the comparison of WiFi and LoRA Module in terms of annual energy consumption. It has been observed from the figure that the annual consumption of WiFi module is more than 2500 Joules. Whereas in LoRA it is up to 500 Jouals. It is further reducedwith 200 Joules only with the association of conditional flag-based machine learning algorithm.Further, the energy efficiency is directly related with the processing time of machine learning algorithms to save energy. As per figure 4.17, the processing time of decision tree is 3.77 Second having accuracy 96% and the random forest consumes 7.58 second with accuracy level of 97%.

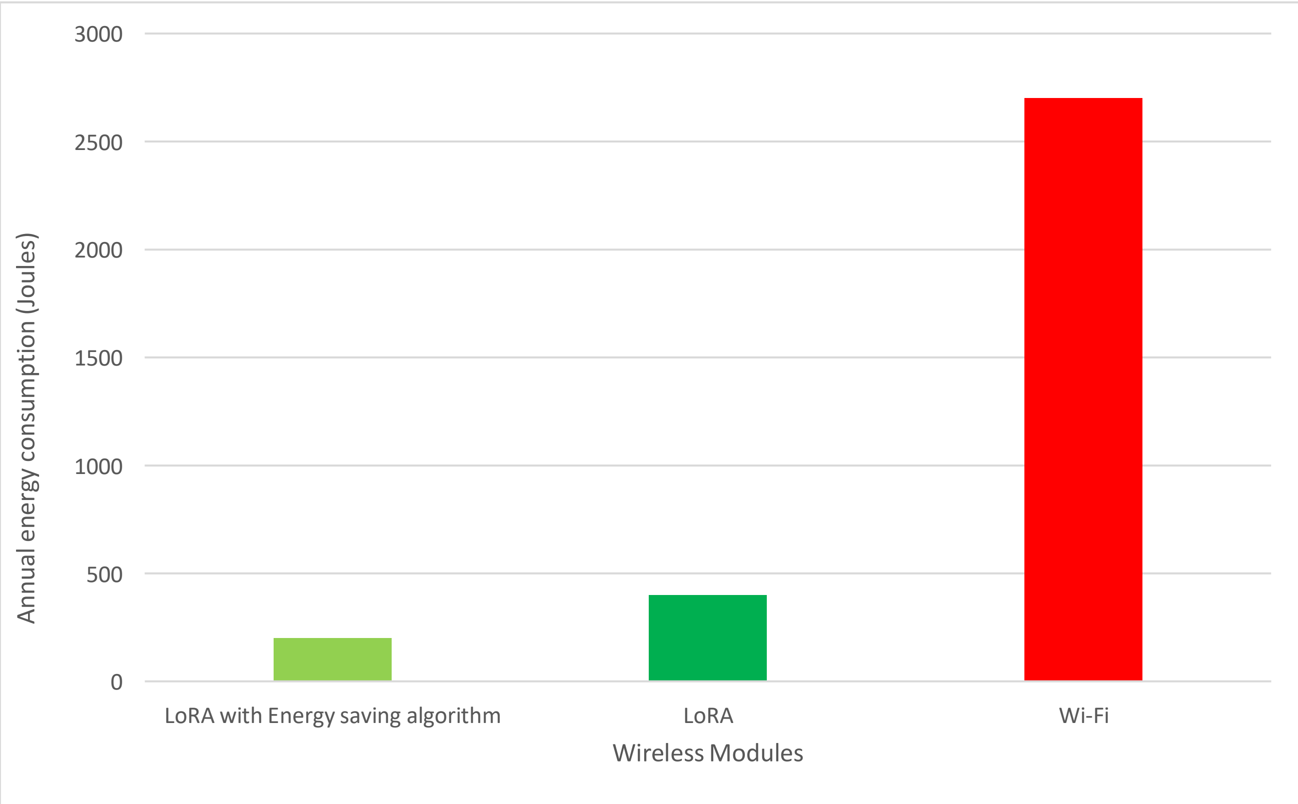

Figure 4.16: Comparison of annual energy consumption by wireless modules (LoRA andWiFi)

The SVM have also achieved 96.89% accuracy by consuming 4 minutes to train himself for classification. As per results the Random Forest algorithm was found best as per requirement of energy saving parameters. The LoRA module is taking very less amount of energy annually to transmit the data from the modules. The Edge computing technique is observed as one of the key factors in achievement of such a huge difference.

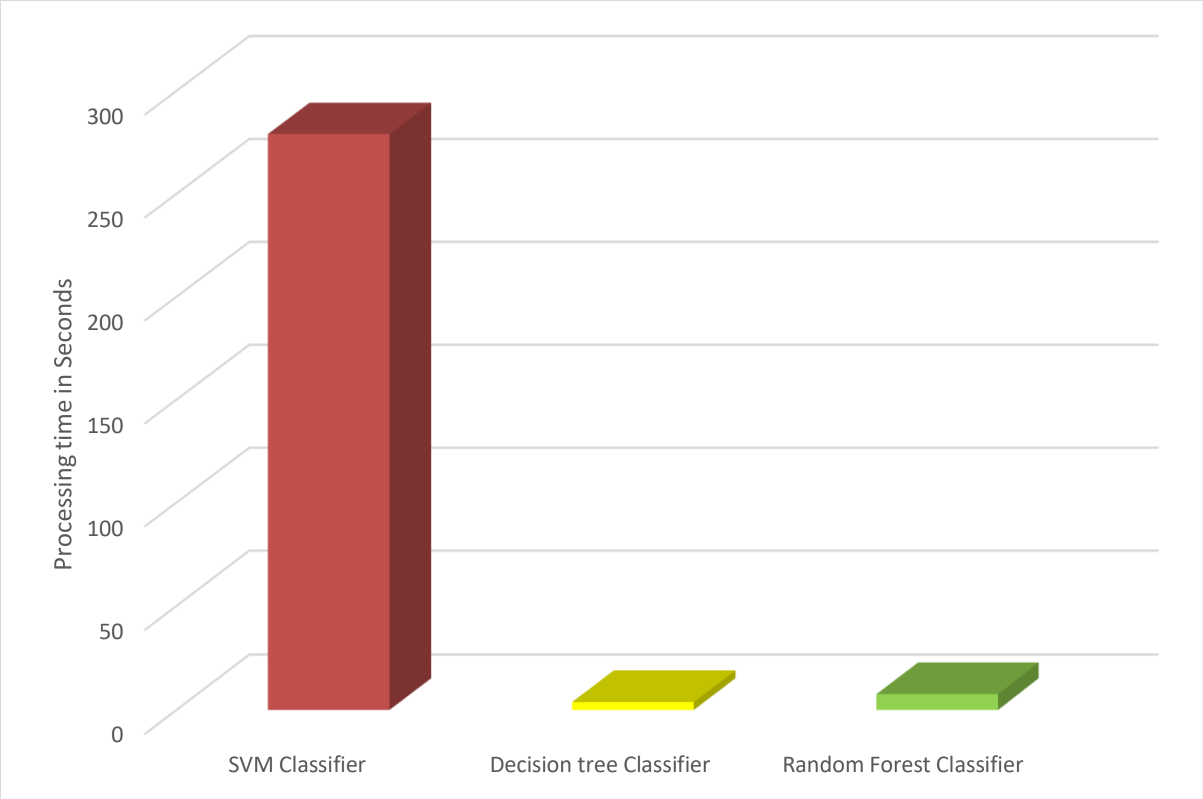

Figure 4.17: Processing time of different classifiers in relation to energy consumption

4.4.1 Comparative analysis of energy efficiency improvement in relation to Job scheduling framework

The growing prominence of Edge of Things (EoT) applications, which represent a convergence of edge computing and Internet of Things (IoT) technologies, has made energy efficiency and consumption critically important. In these applications, numerous sensors having limited power resources, are deployed in various environments to collect, process, and transmit data. The energy demands of these devices can be substantial, especially in scenarios where they are expected to operate continuously over extended periods, often in remote or hard-to-reach locations[115]. Efficient energy use in EoT applications is not just a matter of prolonging battery life but it is also about reducing operational costs, minimizing maintenance needs, and enhancing the sustainability of the technology. For instance, a sensor network monitoring environmental parameters in a remote area must be energy efficient to reduce the frequency of battery replacements, which could be logistically challenging and costly. Similarly, in urban settings, EoT devices that optimize energy consumption contribute to reducing the overall energy footprint of smart city infrastructure. Energy efficiency in EoT applications has a direct impact on the scalability and feasibility of these technologies. As the number of connected devices continues to soar, the cumulative energy demand could become significant. Efficient energy use helps in mitigating this demand, thus supporting the sustainable growth of EoT networks[207]. The focus on energy efficiency is crucial for ensuring the practicality, cost-effectiveness, and environmental friendliness of Edge of Things applications, making it a key consideration in their design and deployment. In this study, we performed two experiments to observe the energy consumption over time and implement efficient job scheduling framework to sustain it for longer time[140].

Table 4.9: Energy consumption of sensors and other modules over the time

S. No	Name of Component	Voltage	Current Rating	Power (Watt)
1	Heart Rate Sensor (MIR 910610)	5 Volt	2mA	0.01 W
2	Blood Pressure sensor (HEM 7600T)	3 Volt	2mA	0.006 W
3	SpO_2 Sensor (Nellcor DS100A)	3 Volt	2mA	0.006 W
4	Blood Sugar Level Sensor (Free Style Libre_ 2040011304)	1.5 Volt	0.5 mA	0.0007 W
5	Accelerometer (ADXL345 3-Axis Digital Accelerometer)	2.5 Volt	10µA	0.000025 W
6	Distance Sensor (HC-SR04)	3 Volt	5mA	0.015 W
7	Microcontroller Module (Raspberry Pi Zero)	5 Volt	0.5 mA	0.0025 W
8	LoRA (915MHz Shield-TTGO T-SIM7000G)	3.3 Volt	2 mA	0.006 W

The specification of components in relation to their power consumption is given below with reference to a 9 Volt battery source having 19440 J capacity to deliver the required power to the components. With reference to equations 3.1 to 3.11, Table 4.9 is showing the specification of components in terms of their demand of voltage and current consumption to measure the specific health related parameter. The energy consumption of module is depending on the amount of energy in form of joules provide by the battery in respect to power consumption by the components over the time. Here the active mode and sleep mode of modules varies with the probability of events to happen over the span of time[72]. The energy efficiency is related to battery discharge over the time which is calculated by equation 4.1.

$$\text{Discharge time} = \frac{\text{Capacity of Battery (Joules)}}{\text{Accumulative power of components and modules over the time (Watt)}} \quad (4.1)$$

Here, the time is considering the activation period, sleep period, processing time, clock cycles to execute the result in microcontroller, transmission time of LoRA module. As per the accumulated power consumption of the components the two experiments were conducted. In the first experiment, the general energy efficiency in terms of battery discharge is calculated and then experiment two was conducted as per Job scheduling framework.

Experiment 1: Standard Job Scheduling (SJS)

Energy Efficiency without Job Scheduling" is a concept that emphasizes the importance of optimizing energy consumption in various systems and processes, but without the reliance on job scheduling techniques. Job scheduling, often used in computing and industrial contexts, involves organizing and prioritizing tasks to optimize resource utilization and operational efficiency.

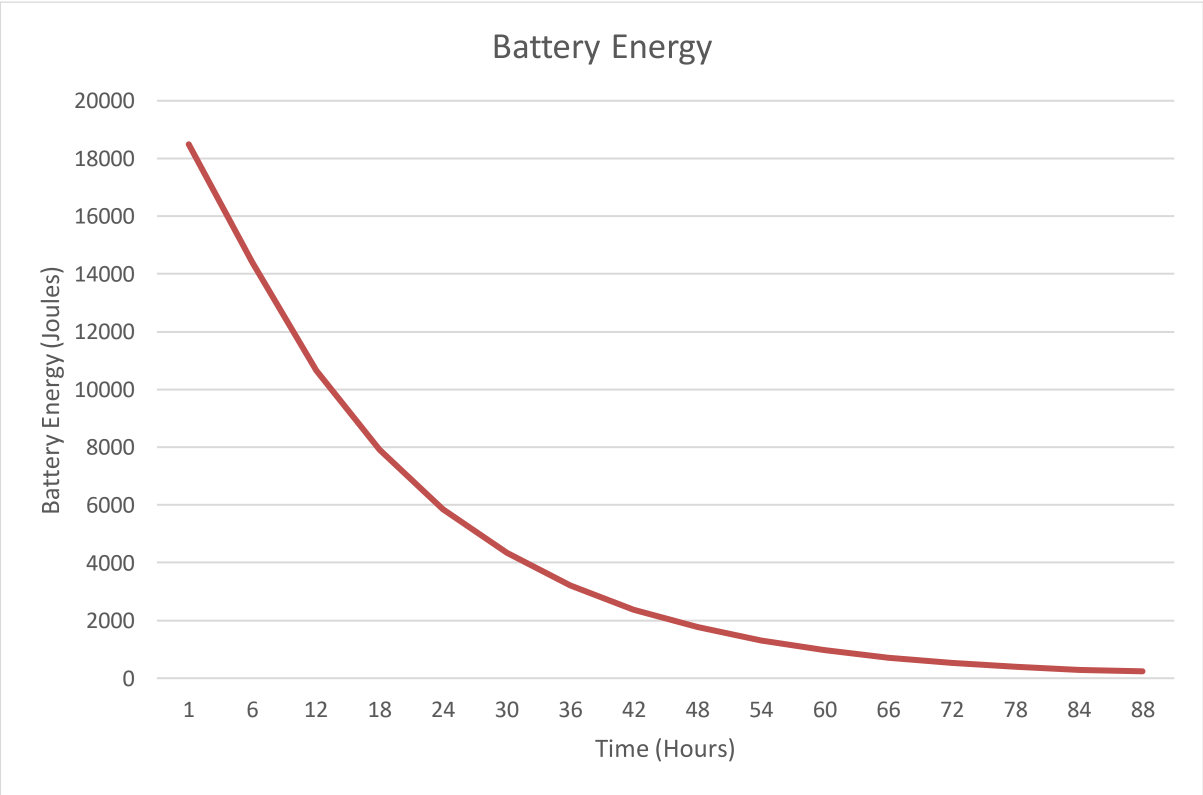

Figure 4.18: Energy efficiency of battery without Job scheduling Framework in IoT based health care monitoring system

However, this approach can sometimes be complex, requiring advanced algorithms and significant management overhead. In this experiment, without job scheduling, the array of sensors was active for n number of time to continually monitor the health parameters from a person under monitoring and then with the observant process of embedded system and LoRA module it will transmit the decision to usen end node. During this process, it has been observed that the LoRa and other modules will consume the power but with the training of AI algorithms it will consume the bunch of power. Figure 4.18 is showing the discharge time of battery over the time without Job scheduling. From the figure 4.18, it can be observed that the sensor modules and AI based training and execution of algorithms continually consumes the bunch of energy over the time.

The battery will take 78 to 82 hours to sustain in the working arena. There is seamless demand of re boosting the energy consumption over the time to sustain the battery even for longer time. The embedded system and wireless transmission consume more power than other sensor modules. As per studies, the LoRA module consumes approximately 10 mA of current to send 10 bytes of packet of data to the designated network and in sleep mode it takes only few micro amperes current to be in the working mode. Keeping in mind the scenario of the nodes, the experiment 2 has been performed to improve the efficiency of connected battery.

Experiment 2: Energy Efficient Job Scheduling (E^2 JS)

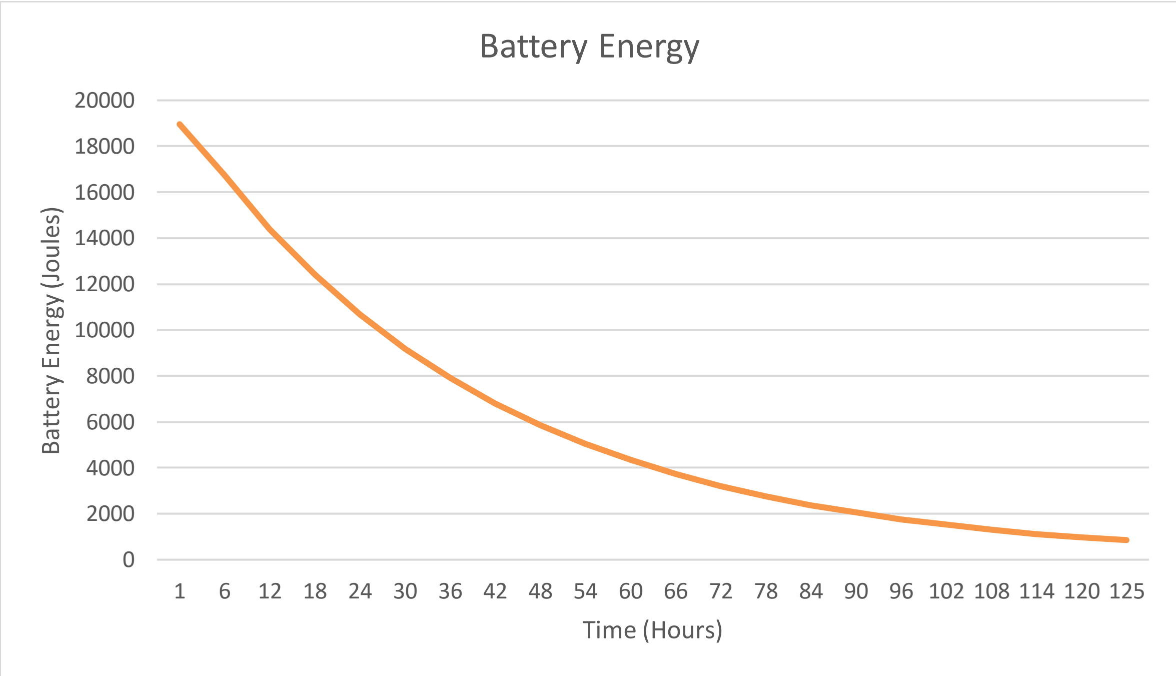

Figure 4.19: Energy efficiency of battery with Job scheduling Framework in IoT based health care monitoring system

As per collected data from the medical experts, the six vital parameters of patients were measured by concerned sensors. The rank of these sensor data was determined to see the importance of these vital parameters to feature the Job scheduling as per rank of parameters. As per results in Figure 4.11, the blood pressure sensor was set to higher priority to observe the condition of a patient as per rank the Heart Rate variability sensor was ranked second to simulate the condition to the user end and then in a consequence, if the abnormality will be detected in the health parameters then the array of sensor will activate and then further activate the connected embedded system to wake up and start working in active mode for indication of health condition to medical expert. As per the probability of detection of abnormality in the concerned patient, figure 4.19 is showing the energy efficiency of battery in relation to job scheduling framework.

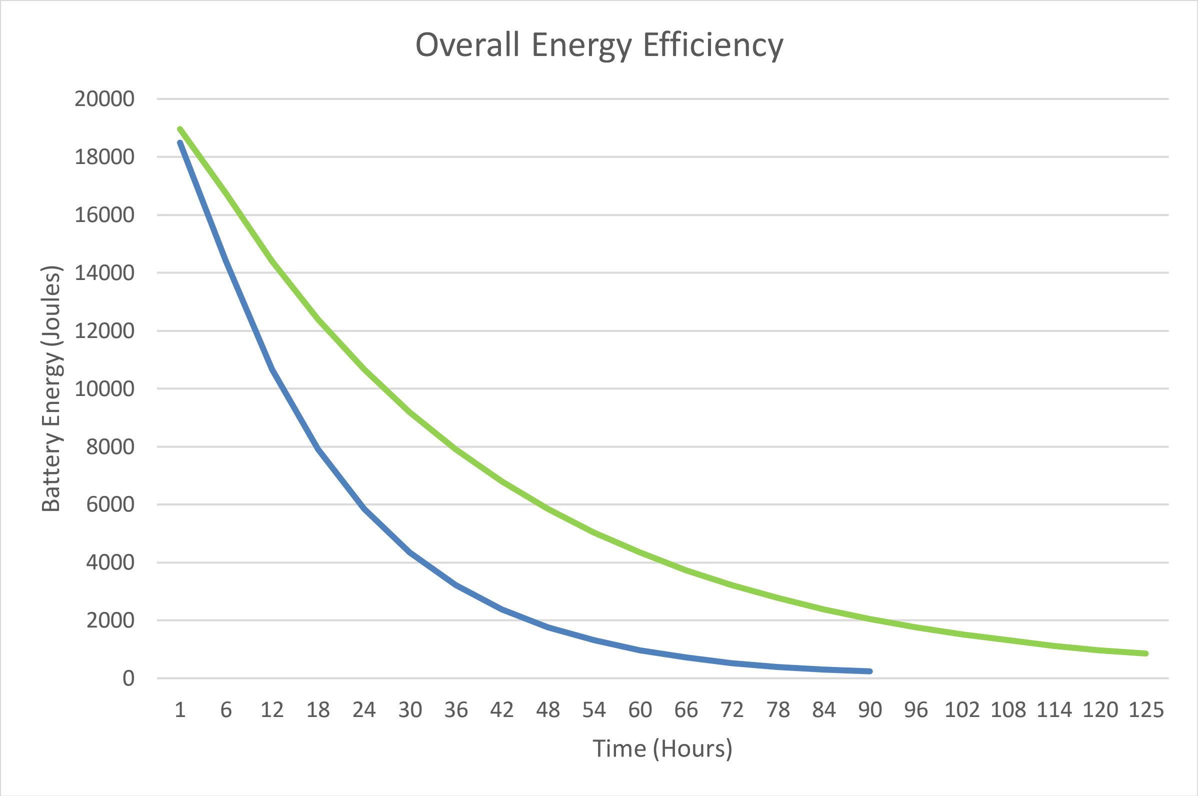

Figure 4.20: Comparative analysis of experiment 1 and 2 in terms of improvement in energy efficiency improvement

The data depicted in Figure 4.19 demonstrates the efficacy of prioritizing sensor activation according to the detection of anomalies. This strategic method effectively preserves battery life by activating only the appropriate nodes in the sensor array network when abnormal conditions are detected. By limiting the duration of sensor operation under regular monitoring settings, the energy consumption is significantly decreased, resulting in a longer battery life. Significantly, in Experiment 2, the implementation of this focused activation approach resulted in the battery's endurance exceeding 115 hours, a considerable improvement compared to Experiment 1. This can be attributable to the improved task scheduling algorithm, which increases energy efficiency by up to 30%. It ensures that devices that require a lot of energy, such as embedded systems and LoRA transmission modules, are only active when there is an aberrant patient condition. This conditional activation has the potential to result in energy savings ranging from 10% to 20% compared to situations where these systems would otherwise remain engaged without any need.

Figure 4.20 presents a detailed comparison of the energy efficiencies obtained in both studies, thereby offering a clearer understanding of these findings. This graphic representation emphasizes the substantial enhancements achieved by optimizing task scheduling and deploying sensors strategically. It highlights the practical advantages of this technique in improving the sustainability and operational lifespan of patient monitoring systems.

Table 4.10: Comparative energy consumption of SJS and E^2JS Systems over various time intervals

Time Hours)	6	12	24	48	96	125
SJS (Joules)	14000	11000	6000	2000	0	0
E^2JS (Joules)	17000	14500	10000	6000	2000	900

As per table 4.10, in the first six hours of time, the battery energy was at level of 14000 Joules with respect to SJS but in case of E2JS, the battery was sustained at 17000 Joules. In the next couples of hours (after 24 Hours) the battery drainage level decreased to 6000 Joules due to SJS but in case of E2JS, the battery level was at 10000 Joules. In consecutive time intervals, due to SJS the battery was full drained to 0 (at 125 Hours from beginning), but in case of E2JS, the battery still contains, 900 Joules. So, overall, the E2JS framework helps to reduce the energy consumption of battery while maintaining the system to monitor the health of a patient.

As per Ham G et al., (2023) the edge computing is one of the leading research areas where there is need of such algorithms who can directly improve the energy management of module with maintaining the performance of other parameters such as range and processing speed (latency). In this study, a very smart and sophisticated strategy was planned to provide care to patients under medical observation. Varies studies used the expensive and continuous processing modules for patient care but they had certain drawbacks[217].

According to study of Zhang J et al., (2023) it has been observed that there is always a compromising tradeoff between the range, power consumption and latency parameters while establishment of data monitoring applications. As per reports, it was clearly observed that annual consumption of energy by the wireless modules somehow a challenging task for research communities to manage for long-term battery- operated tasks[41].
In this study, various wireless modules were explored with respect to their specifications but

LoRA module was used for establishment of communication between sensor data module and data processing module in a remote situation. From the figure 4.16 to 4.20, it is very clear that the energy saving framework helps to enhance 30% of energy usage than the experiment 1.

The activation of concerned sensors and modules as per rank of parameter supports the decisions take by machine learning algorithms. As per study of Truong V et. al., (2021) it has been observed that the WiFi module is not able to operate for long time due to its high capacity of power consumption than other wireless modules such as Zigbee and LoRA. However, a tradeoff between speed and latency is always matter of concern in these wireless modules. LoRA module can cover a long range of distance and have capability of sufficient transmission time capability to overcome the problem of latency in real time processing[218]. In this study, the LoRA module takes 45 to 55 milliseconds of time to transceiver the data. As per results, the latency effect was null and the LoRA module consumes very less amount of energy than WiFi (Figure: 4.16).

From the results it has been observed that the adoption of AI based smart control strategy is also successfully reduced the power consumption. The adoption of machine learning algorithms can prioritize the tasks such as whether to do or not. In this study, the data collection module was switched to on and off mode based on flag. As the decision tree and random forest algorithms were trained with six ensemble-based attributes and based on them the situation was also trained and tested during training period. The machine learning algorithms were able to categorize the decision based on normal and abnormal situation of a patient. Both algorithms were governing the LoRA receiver to send the signal to LoRA transmitted to whether there is need to collect the data or not. The periodic signaling related to situation and activation of real time data collection process, reduces the demand of power up to some extent. As per real time scenario, the classification methods performed with more than 95% accuracy and the F1 score was also more than 95.5 %.

It has been observed from the studies that algorithms are very competitive to classify the situation in respect of energy consumption with very short span of time such as in millisecond to microlevel once get trained by the data. It has been observed from the studies that the random forest has better computation power in terms of overfitting than the decision tree and SVM for prediction of situation. The probabilistic distance between them is varied from 0.05

to 0.08. This is also depending on the measured data and the decision based on measured attributes. From the results, it has been observed that out of 10 real time input situations given at a time to both trained algorithms, the random forest has 97% success rate success rate in the prediction of outcomes in relation to true values. In support to other studies, it is very clear that the number of inputs and their correlation with each other supports the decision classifier to categories the final decision.

4.5 Conclusion

For patient monitoring under certain situations, it is very important to measure the data with inter relatively for building a relative decision according to the situation. In this study, the data of more than 40,000 reading from selected human subjects (under medical monitoring) were collected and processed for training of machine learning algorithms. As per reports, the amount of data was enough in each attribute for learning of AI algorithms. Hence, the more than 96% accuracy was achieved in this study. As per results, the LoRA with energy efficient algorithms plays a very important role in development of edge based medical applications such as from monitoring to treatment of patients. By these methods, the real time processing and decision taking capability of machines can be improved on local servers and need of very powerful infrastructures may be reduced. Our study is very well supported by the results of similar kind of research reports.

Chapter 5

Conclusion and future scope

5.1 Conclusion

In the era of computing, edge computing is one of the revolutionary techniques through which the solution of the situation can be estimated within fraction of seconds at the spot. Earlier, the information was first collected and then it was loaded to cloud for observation of decision to be given as per condition from a remote server. During this process various issues were observed related to cost, computation power, latency, energy consumption etc. The research communities also looking this challenge as opportunity to solve such issues. The edge computing is one of the efficient alternate to solve the challenges. In edge computing hierarchy, the data has been collected at the end user node and the decision will be calculated at the node itself rather than sending it to far clouds. By this way, various key parameters improved. As per literature survey, the edge computing plays a key role in accelerating the development of new applications in diverse fields. As per current scenario, there are various fields such as agriculture, automation industry, automotive industry, and health sector where there is need of an hour to identify the challenges and propose the solution to end user. The health sector has his own importance in relation to mankind. According to medical experts, 75% of casualties can be prevented if the information of vital organs can be measured, diagnosed, and cured at the spot. In hospitals, it is one of the challenges to monitor the patient continually for his or her care. There are various physiological parameters such as heart rate, blood pressure, sugar level etc. which needs rigorous monitoring to observe the condition of a patient under various positions.

In this study, a smart edge computing-based health monitoring system has been developed to facilitate the medical team and patient under observation. In the initial stage of this study, the data of patients under medical observation was collected from the medical expert. The medical expert observed six key parameters from the patients such as blood pressure, Heart Rate Variability, Sugar level, Oxygen Saturation level (SpO2), Position in form of acceleration and his distance from the observation module and then based on the observation they noted the situation of patient in three categories such as No Fall detected, Slip detected, Definite fall. In this study, more than 40,000 situations were used to train the classification algorithm. The measured features were observed to validate the data. From the box plot and histogram, the treads in features were observed. Afterwords, the correlation matrix, and rank graphs were plotted to understand the significance and contribution of features in relation to classification of decision. From the results, it has been observed that the blood pressure and heart rate variability is ranked highest to observe the immediate situation of a patient under medical surveillance followed by rest features. As per the objective of this study, a smart monitoring strategy has been prepared to measure the prime features and then as per observation of abnormality, the subsequent sensors were activated to measure other features. As per situation the node will decide to capture the data and then with the help of machine learning, the information will be passed to medical team for necessary action. In this study, as per aim of saving the energy, a smart energy management framework is developed who is supported by rank wise activation and deactivation of sensor nodes and other integrated modules.

The LoRA module was adopted to establish the wireless communication between the end node and master node. The LoRA module takes 45 to 55 milliseconds of time to transceiver the data. In this study, various classification methods were tested but random forest and decision tree classification methods were found to be suitable for prediction of situation due to their response time and accuracy with the amount of data compared to SVM classifier. As per computation /processing time based comparative analysis of these algorithms, the Decision tree and random forest algorithms took only few seconds to train whereas the SVM took couple of minutes to generate the decision on the situation. Hence the energy consumption is more in SVM than Random Forest and decision tree algorithm. In this study, classifiers were trained and tested with the collected data and then tested with real time data. From the comparative results, it has been observed that the 90% to 95% accuracy was achieved in relation to actual and predicted situation of a patient. In terms of energy saving process, the job scheduling algorithm helps to save energy in terms of sustainability of battery for 30 % more hours than in experiment 1 (Figure 4.20). In chapter four, the results are very well justified with previous studies.

5.2 Future Scope

Our research provides new opportunities for the academic and research community to investigate and improve the processing of features for decision-making in categorized challenges. Despite technological progress, there is still a significant lack of intelligence in the design of job scheduling frameworks that aim to optimize energy efficiency. The utilization of machine learning techniques in this particular situation showcases notable adaptability, enabling seamless incorporation into various procedures for both regression and classification tasks.

In the future, it is important to determine certain situations where the ranking of algorithms, considering their effectiveness and applicability to the current task, can be flexibly applied using edge computing technologies. Implementing this strategic deployment will allow for more advanced and situation-sensitive processing, resulting in a decrease in the overall expenses and resources needed. Future research should prioritize the development of adaptive and intelligent systems capable of leveraging real-time data to autonomously enhance efficiency and performance.

In addition, utilizing edge computing can enhance the decentralized processing of data, which is especially advantageous for real-time applications that necessitate prompt computational replies. By incorporating more intelligent algorithms and machine learning models that can function efficiently in the periphery of the network, these systems can achieve greater autonomy, minimizing the need for central processing units and thereby saving energy and bandwidth. This method offers not only improvements in operational efficiency but also enables the implementation of more sustainable and cost-effective solutions in many industries.

List of Publications

[1] R. Kanday and R. Singh, "Smart Framework for Energy-Efficient Workload Scheduling for IoT-Based Healthcare Devices," *African J. Biol. Sci. (South Africa)*, vol. 6, pp. 1049–1068, 2024, doi: 10.33472/AFJBS.6.Si2.2024.1049-1068.

[2] R. Kanday and R. Singh, "Implementing an Intelligent Energy Management Framework for IoT Edge Devices," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering, UPCON 2023*, School of Computer Science and Engineering, Lovely Professional University, Punjab, 144411, India: Institute of Electrical and Electronics Engineers Inc., 2023, pp. 47–51. doi: 10.1109/UPCON59197.2023.10434829.

[3] R. Kanday and R. Singh, "Edge Computing for Sustainable Energy Management: An Environmental Aspect," in *Proceedings of International Conference on Contemporary Computing and Informatics, IC3I 2023*, School of Computer Science and Engineering, Lovely Professional University, Punjab, India: Institute of Electrical and Electronics Engineers Inc., 2023, pp. 6–12. doi: 10.1109/IC3I59117.2023.10398093.

[4] R. Kanday, Y. Pimpale, and S. Gupta, "Smart Biomedical Devices- Smart Alcohol Detection System With Engine Cutoff and Notifier," in *AIP Conference Proceedings*, S. V., V. M., P. P.K., and K. R., Eds., School of Computer Sciences and Engineering, Lovely Professional University, Punjab, Phagwara, India: American Institute of Physics Inc., 2023. doi: 10.1063/5.0163484.

[5] Y. Pimpale, S. Gupta, and R. Kanday, "Toxicity of Arsenic and its Effect in Asian Villages Due to Use of Dunk Cakes as Domestic Fuel Using Artificial Neural Network," in *AIP Conference Proceedings*, S. V., V. M., P. P.K., and K. R., Eds., Lovely Professional University, Punjab, Phagwara, 144411, India: American Institute of Physics Inc., 2023. doi: 10.1063/5.0163483.

[6] M. K. Rai, R. Kanday, and R. Thomas, "Current Issues and Challenges of Security in IoT Based Applications," in *Advances in Intelligent Systems and Computing*, X. F., P. S., and T. M., Eds., School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, 144411, Punjab, India: Springer, 2020, pp. 578–589. doi: 10.1007/978-3-030-34387-3_71.

[7] R. Kanday, "A survey on cloud computing security," in *Proceedings: Turing 100 - International Conference on Computing Sciences, ICCS 2012*, Department of Computer Science and Applications, Lovely Professional University, Phagwara, India, 2012, pp. 302–311. doi: 10.1109/ICCS.2012.6.

References

[1] J. Zhang et al., “Dependent task offloading mechanism for cloud–edge-device collaboration,” Journal of Network and Computer Applications, vol. 216, 2023, doi: 10.1016/j.jnca.2023.103656.

[2] B. Hu, Z. Yan, and M. Zhao, “Workload-Aware Scheduling of Multiple- Criticality Real-Time Applications in Vehicular Edge Computing System,” IEEE Trans Industr Inform, vol. 19, no. 10, pp. 10091–10101, 2023, doi: 10.1109/TII.2022.3231420.

[3] K. Röbert, H. Bornholdt, M. Fischer, and J. Edinger, “Latency-Aware Scheduling for Real-Time Application Support in Edge Computing,” in EdgeSys 2023 - Proceedings of the 6th International Workshop on Edge Systems, Analytics and Networking, Part of EuroSys 2023, 2023, pp. 13–18. doi: 10.1145/3578354.3592866.

[4] E. Sisinni et al., “LoRaWAN Range Extender for Industrial IoT,” IEEE Trans Industr Inform, vol. 16, no. 8, pp. 5607–5616, 2020, doi: 10.1109/TII.2019.2958620.

[5] I. Butun, Industrial IoT: Challenges, design principles, applications, and security. 2020.

[6] Q. Xie, C. Zhang, and X. Jia, “Security-Aware and Efficient Data Deduplication for Edge-Assisted Cloud Storage Systems,” IEEE Trans Serv Comput, vol. 16, no. 3, pp. 2191–2202, 2023, doi: 10.1109/TSC.2022.3195318.

[7] P. De Moraes and A. F. Da Conceicao, “Protecting LoRaWan data against untrusted network servers,” in Proceedings - IEEE Congress on Cybermatics: 2021 IEEE International Conferences on Internet of Things, iThings 2021, IEEE Green Computing and Communications, GreenCom 2021, IEEE Cyber, Physical and Social Computing, CPSCom 2021 and IEEE Smart Data, SmartD, 2021, pp. 99–106. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics53846.2021.00029.

[8] R. McPherson and J. Irvine, “Secure Decentralised Deployment of LoRaWAN Sensors,” IEEE Sens J, vol. 21, no. 1, pp. 725–732, 2021, doi: 10.1109/JSEN.2020.3013117.

[9] N. E. H. Djidi, M. Gautier, A. Courtay, O. Berder, and M. Magno, “How can wake-up radio reduce lora downlink latency for energy harvesting sensor nodes?,” Sensors (Switzerland), vol. 21, no. 3, pp. 1–16, 2021, doi: 10.3390/s21030733.

[10] A. Bosio, L. Dolecek, A. Kourfali, S. Parameswaran, and A. Savino, “Special Issue: ‘Approximation at the Edge,’” ACM Transactions on Embedded Computing Systems, vol. 22, no. 4, 2023, doi: 10.1145/3605757.

[11] R. Fereira, C. Ranaweera, K. Lee, and J.-G. Schneider, “Energy Efficient Node Selection in Edge-Fog-Cloud Layered IoT Architecture,” Sensors, vol. 23, no. 13, 2023, doi: 10.3390/s23136039.

[12] S. S. Hoseini, T. Beyer, A. Ghaderi, and Z. Movahedi, “Latency-aware SDN-based Mobile Edge Computation Offloading in Industrial IoT,” in 2023 28th International

Computer Conference, Computer Society of Iran, CSICC 2023, 2023. doi: 10.1109/CSICC58665.2023.10105378.

[13] J. Lim, "Latency-Aware Task Scheduling for IoT Applications Based on Artificial Intelligence with Partitioning in Small-Scale Fog Computing Environments," Sensors, vol. 22, no. 19, 2022, doi: 10.3390/s22197326.

[14] C.-L. Chen, C. G. Brinton, and V. Aggarwal, "Latency Minimization for Mobile Edge Computing Networks," IEEE Trans Mob Comput, vol. 22, no. 4, pp. 2233–2247, 2023, doi: 10.1109/TMC.2021.3117511.

[15] M. Tajallifar, S. Ebrahimi, M. R. Javan, N. Mokari, and L. Chiaraviglio, "Energy-Efficient Task Offloading under E2E Latency Constraints," IEEE Transactions on Communications, vol. 70, no. 3, pp. 1711–1725, 2022, doi: 10.1109/TCOMM.2021.3132909.

[16] M. Morsali, M. Nazzal, A. Khreishah, and S. Angizi, "IMA-GNN: In-Memory Acceleration of Centralized and Decentralized Graph Neural Networks at the Edge," in Proceedings of the ACM Great Lakes Symposium on VLSI, GLSVLSI, 2023, pp. 3–8. doi: 10.1145/3583781.3590248.

[17] M. Mihalik, M. Hrubos, and E. Bubenikova, "Wireless proximity sensor in lorawan network," in International Conference on Applied Electronics, 2020. doi: 10.23919/AE49394.2020.9232725.

[18] V. Sharma and A. Roy, "Localized Indoor / Outdoor Lora Net Work A Review," in 2023 1st International Conference on Intelligent Computing and Research Trends, ICRT 2023, 2023. doi: 10.1109/ICRT57042.2023.10146662.

[19] B. Su, Z. Qin, and Q. Ni, "Energy Efficient Uplink Transmissions in LoRa Networks," IEEE Transactions on Communications, vol. 68, no. 8, pp. 4960–4972, 2020, doi: 10.1109/TCOMM.2020.2993085.

[20] Y. Sun, W. Jiang, Y. Yang, H. Zhu, and Y. Jiang, "Multi-domain authorization and decision-making method of access control in the edge environment," Computer Networks, vol. 228, 2023, doi: 10.1016/j.comnet.2023.109721.

[21] F. S. Dantas Silva et al., "A survey on long-range wide-area network technology optimizations," IEEE Access, vol. 9, pp. 106079–106106, 2021, doi: 10.1109/ACCESS.2021.3079095.

[22] B. Hu, Z. Yan, and M. Zhao, "Workload-Aware Scheduling of Multiple- Criticality Real-Time Applications in Vehicular Edge Computing System," IEEE Trans Industr Inform, vol. 19, no. 10, pp. 10091–10101, 2023, doi: 10.1109/TII.2022.3231420.

[23] W. Zhang, Z. Zhang, S. Zeadally, H.-C. Chao, and V. C. M. Leung, "Energy-efficient workload allocation and computation resource configuration in distributed cloud/edge computing systems with stochastic workloads," IEEE Journal on Selected Areas in Communications, vol. 38, no. 6, pp. 1118–1132, 2020, doi: 10.1109/JSAC.2020.2986614.

[24] Y. Zhao, Z. Yang, X. He, X. Cai, X. Miao, and Q. Ma, "Trine: Cloud-Edge-Device Cooperated Real-Time Video Analysis for Household Applications," IEEE Trans Mob Comput, vol. 22, no. 8, pp. 4973–4985, 2023, doi: 10.1109/TMC.2022.3154721.

[25] P. M. Subramanian and A. Rajeswari, LoRa-Based Infrastructure for Medical IoT System. 2021. doi: 10.1007/978-981-16-1769-0_30.

[26] K. Thirunavukkarasu, L. Raju, and S. Sathishbabu, "A Survey on LoRaWAN for Smart Medical and Industries," in Proceedings of the 9th International Conference on Electrical Energy Systems, ICEES 2023, 2023, pp. 34–40. doi: 10.1109/ICEES57979.2023.10110099.

[27] D. G. Lema, R. Usamentiaga, and D. F. García, "Low-cost system for real-time verification of personal protective equipment in industrial facilities using edge computing devices," J Real Time Image Process, vol. 20, no. 6, 2023, doi: 10.1007/s11554-023-01368-7.

[28] V. Joshi and K. Patil, "Delay-Energy Aware Task Offloading and VM Migration Policy for Mobile Edge Computing," Wirel Pers Commun, vol. 123, no. 4, pp. 3309–3326, 2022, doi: 10.1007/s11277-021-09290-6.

[29] W. Su et al., "Cloud-Edge Computing-Based ICICOS Framework for Industrial Automation and Artificial Intelligence: A Survey," Journal of Circuits, Systems and Computers, vol. 32, no. 10, 2023, doi: 10.1142/S0218126623501682.

[30] Q. Dong, L. Zhao, J. Zhao, and X. Liu, "Synthesis Decision Approach of LORA Based on Decision Flow," in Proceedings of 2020 IEEE International Conference on Artificial Intelligence and Information Systems, ICAIIS 2020, 2020, pp. 360–365. doi: 10.1109/ICAIIS49377.2020.9194873.

[31] A. Shirke and M. M. Chandane, "Collaborative offloading decision policy framework in IoT using edge computing," Multimed Tools Appl, 2023, doi: 10.1007/s11042-023-14383-4.

[32] D. P. Mtowe and D. M. Kim, "Edge-Computing-Enabled Low-Latency Communication for a Wireless Networked Control System," Electronics (Switzerland), vol. 12, no. 14, 2023, doi: 10.3390/electronics12143181.

[33] D. P. Mtowe and D. M. Kim, "Edge-Computing-Enabled Low-Latency Communication for a Wireless Networked Control System," Electronics (Switzerland), vol. 12, no. 14, 2023, doi: 10.3390/electronics12143181.

[34] D. Shekar Goud et al., "Deep learning technique for patients healthcare monitoring using IoT body based body sensors and edge servers," Journal of Intelligent and Fuzzy Systems, vol. 45, no. 4, pp. 7161–7175, 2023, doi: 10.3233/JIFS-231239.

[35] A. Ahmed, S. Abdullah, M. Bukhsh, I. Ahmad, and Z. Mushtaq, "An Energy-Efficient Data Aggregation Mechanism for IoT Secured by Blockchain," IEEE Access, vol. 10, pp. 11404–11419, 2022, doi: 10.1109/ACCESS.2022.3146295.

[36] V. N. Moothedath, J. P. Champati, and J. Gross, “Energy Efficient Sampling Policies for Edge Computing Feedback Systems,” IEEE Trans Mob Comput, vol. 22, no. 8, pp. 4634–4647, 2023, doi: 10.1109/TMC.2022.3165852.

[37] A. Tsakmakis, A. Valkanis, G. Beletsioti, K. Kantelis, P. Nicopolitidis, and G. Papadimitriou, “Efficient Frequency Planning for LoRaWAN networks with urgent traffic,” in Proceedings of the International Conference on Computer, Information, and Telecommunication Systems, CITS 2021, 2021. doi: 10.1109/CITS52676.2021.9618355.

[38] K. Lasri, Y. Ben Maissa, L. Echabbi, O. Iova, and F. Valois, “Probabilistic and distributed traffic control in LPWANs,” Ad Hoc Networks, vol. 143, 2023, doi: 10.1016/j.adhoc.2023.103121.

[39] S. R. Gopal and V. S. V. Prabhakar, “A hybrid approach to enhance scalability, reliability and computational speed in LoRa networks,” International Journal of Advanced Computer Science and Applications, vol. 11, no. 9, pp. 463–469, 2020, doi: 10.14569/IJACSA.2020.0110955.

[40] A. Xu, W. Hou, Y. Zhang, Y. Jiang, W. Lei, and H. Wen, A Novel Smart Grid Model for Efficient Resources, vol. 1084 AISC. 2020. doi: 10.1007/978-3-030-34387-3_101.

[41] J. Zhang, J. Wang, Z. Yuan, W. Zhang, and L. Liu, “Offloading Demand Prediction-Driven Latency-Aware Resource Reservation in Edge Networks,” IEEE Internet Things J, vol. 10, no. 15, pp. 13826–13836, 2023, doi: 10.1109/JIOT.2023.3263188.

[42] X. Li, T. Zhang, S. Wang, G. Zhu, R. Wang, and T.-H. Chang, “Large-Scale Bandwidth and Power Optimization for Multi-Modal Edge Intelligence Autonomous Driving,” IEEE Wireless Communications Letters, vol. 12, no. 6, pp. 1096–1100, 2023, doi: 10.1109/LWC.2023.3262573.

[43] S. Song, S. Ma, X. Zhu, Y. Li, F. Yang, and L. Zhai, “Joint bandwidth allocation and task offloading in multi-access edge computing,” Expert Syst Appl, vol. 217, 2023, doi: 10.1016/j.eswa.2023.119563.

[44] F. Valente, V. Eramo, and F. G. Lavacca, “Optimal bandwidth and computing resource allocation in low earth orbit satellite constellation for earth observation applications,” Computer Networks, vol. 232, 2023, doi: 10.1016/j.comnet.2023.109849.

[45] M. H. Kashani and E. Mahdipour, “Load Balancing Algorithms in Fog Computing,” IEEE Trans Serv Comput, vol. 16, no. 2, pp. 1505–1521, 2023, doi: 10.1109/TSC.2022.3174475.

[46] V. Gowri and B. Baranidharan, Dynamic Energy Efficient Load Balancing Approach in Fog Computing Environment, vol. 131. 2023. doi: 10.1007/978-981-19-1844-5_13.

[47] W. Liu, H. Zhang, H. Ding, and D. Yuan, “Delay and Energy Minimization for Adaptive Video Streaming: A Joint Edge Caching, Computing and Power Allocation Approach,” IEEE Trans Veh Technol, vol. 71, no. 9, pp. 9602–9612, 2022, doi: 10.1109/TVT.2022.3179696.

[48] C. Buratti et al., IoT protocols, architectures, and applications. 2021. doi: 10.1016/B978-0-12-820581-5.00013-4.

[49] A. Yegin et al., LoRaWAN protocol: specifications, security, and capabilities. 2020. doi: 10.1016/B978-0-12-818880-4.00003-X.

[50] C. Zhong and A. Springer, "Design and evaluation of innovative protocols for LoRa," IET Wireless Sensor Systems, vol. 12, no. 1, pp. 12–20, 2022, doi: 10.1049/wss2.12033.

[51] S. Yang, J. Liu, F. Zhang, F. Li, X. Chen, and X. Fu, "Caching-Enabled Computation Offloading in Multi-Region MEC Network via Deep Reinforcement Learning," IEEE Internet Things J, vol. 9, no. 21, pp. 21086–21098, 2022, doi: 10.1109/JIOT.2022.3176289.

[52] Z. Chen, Z. Chen, Z. Ren, L. Liang, W. Wen, and Y. Jia, "Joint optimization of task caching, computation offloading and resource allocation for mobile edge computing," China Communications, 2022, doi: 10.23919/JCC.2022.00.002.

[53] H. Zhang, Y. Cheng, K. Liu, and X. He, "The Mobility Management Strategies by Integrating Mobile Edge Computing and CDN in Vehicular Networks " Dianzi Yu Xinxi Xuebao/Journal of Electronics and Information Technology, vol. 42, no. 6, pp. 1444–1451, 2020, doi: 10.11999/JEIT190571.

[54] G. Park, W. Lee, and I. Joe, "Network resource optimization with reinforcement learning for low power wide area networks," EURASIP J Wirel Commun Netw, vol. 2020, no. 1, 2020, doi: 10.1186/s13638-020-01783-5.

[55] Y. Lu, X. Chen, Y. Zhang, and Y. Chen, "Cost-Efficient Resources Scheduling for Mobile Edge Computing in Ultra-Dense Networks," IEEE Transactions on Network and Service Management, vol. 19, no. 3, pp. 3163–3173, 2022, doi: 10.1109/TNSM.2022.3163297.

[56] S. George, H. Turki, Z. Feng, D. Ramanan, P. Pillai, and M. Satyanarayanan, "Edge-based Privacy-Sensitive Live Learning for Discovery of Training Data," in NetAISys 2023 - Proceedings of the 1st International Workshop on Networked AI Systems, Part of MobiSys 2023, 2023, pp. 19–24. doi: 10.1145/3597062.3597279.

[57] C.-L. Yang, H. Tampubolon, A. Setyoko, K.-L. Hua, M. Tanveer, and W. Wei, "Secure and Privacy-Preserving Human Interaction Recognition of Pervasive Healthcare Monitoring," IEEE Trans Netw Sci Eng, vol. 10, no. 5, pp. 2439–2454, 2023, doi: 10.1109/TNSE.2022.3223281.

[58] W. Cao, W. Shen, Z. Zhang, and J. Qin, "Privacy-preserving healthcare monitoring for IoT devices under edge computing," Comput Secur, vol. 134, 2023, doi: 10.1016/j.cose.2023.103464.

[59] S. Pandey and A. Kaur, "ENCRYPTION-BASED SECURE FRAMEWORK FOR SMART TRAFFIC MANAGEMENT USING FOG COMPUTING," J Theor Appl Inf Technol, vol. 101, no. 10, pp. 3757–3768, 2023.

[60] K. Prabhavathi, M. B. Anandaraju, and Kiran, “An efficient medical image encryption algorithm for telemedicine applications,” Microprocess Microsyst, vol. 101, 2023, doi: 10.1016/j.micpro.2023.104907.

[61] J. Li, Z. Qin, W. Liu, and X. Yu, “Energy-Aware and Trust-Collaboration Cross-Domain Resource Allocation Algorithm for Edge-Cloud Workflows,” IEEE Internet Things J, 2023, doi: 10.1109/JIOT.2023.3315339.

[62] M. Lapegna, V. Mele, and D. Romano, “Clustering Algorithms for Enhanced Trustworthiness on High-Performance Edge-Computing Devices,” Electronics (Switzerland), vol. 12, no. 7, 2023, doi: 10.3390/electronics12071689.

[63] V. A. Dang, Q. Vu Khanh, V.-H. Nguyen, T. Nguyen, and D. C. Nguyen, “Intelligent Healthcare: Integration of Emerging Technologies and Internet of Things for Humanity,” Sensors, vol. 23, no. 9, 2023, doi: 10.3390/s23094200.

[64] A. R. Yeruva, C. S. L. Vijaya Durga, B. Gokulavasan, K. Pant, P. Chaturvedi, and A. P. Srivastava, “A Smart Healthcare Monitoring System Based on Fog Computing Architecture,” in Proceedings of International Conference on Technological Advancements in Computational Sciences, ICTACS 2022, 2022, pp. 904–909. doi: 10.1109/ICTACS56270.2022.9987881.

[65] V. Kumar, V. Kumar, N. Singh, and R. Kumar, “Enhancing Intrusion Detection System Performance to Detect Attacks on Edge of Things,” SN Comput Sci, vol. 4, no. 6, 2023, doi: 10.1007/s42979-023-02242-w.

[66] Y. Zhong and S. Zhong, “Track Signal Intrusion Detection Method Based on Deep Learning in Cloud-Edge Collaborative Computing Environment,” Journal of Circuits, Systems and Computers, vol. 32, no. 15, 2023, doi: 10.1142/S0218126623502675.

[67] D. Heeger and J. Plusquellic, “Analysis of IoT Authentication over LoRa,” in Proceedings - 16th Annual International Conference on Distributed Computing in Sensor Systems, DCOSS 2020, 2020, pp. 458–465. doi: 10.1109/DCOSS49796.2020.00078.

[68] D. Naidu and N. K. Ray, “Review on Authentication Schemes for Device Security in LoRaWAN,” in Proceedings - 2021 19th OITS International Conference on Information Technology, OCIT 2021, 2021, pp. 387–392. doi: 10.1109/OCIT53463.2021.00082.

[69] C.-I. Fan, E.-S. Zhuang, A. Karati, and C.-H. Su, “A Multiple End-Devices Authentication Scheme for LoRaWAN,” Electronics (Switzerland), vol. 11, no. 5, 2022, doi: 10.3390/electronics11050797.

[70] W. Cao, W. Shen, Z. Zhang, and J. Qin, “Privacy-preserving healthcare monitoring for IoT devices under edge computing,” Comput Secur, vol. 134, 2023, doi: 10.1016/j.cose.2023.103464.

[71] V. Sharma, A. Roy, and T. Raj, “Scalability Analysis and Performance Measurement of LOra Network for Smart City Using Internet of Things,” in Proceedings of the 8th International Conference on Signal Processing and Integrated Networks, SPIN 2021, 2021, pp. 112–117. doi: 10.1109/SPIN52536.2021.9566082.

[72] L.-T. Tu, A. Bradai, Y. Pousset, and A. I. Aravanis, “Energy Efficiency Analysis of LoRa Networks,” IEEE Wireless Communications Letters, vol. 10, no. 9, pp. 1881–1885, 2021, doi: 10.1109/LWC.2021.3084996.

[73] L. Zemko and P. Cicak, “IoT and LPWAN Networks: Increasing Efficiency by Communication Planning,” in 2022 45th International Conference on Telecommunications and Signal Processing, TSP 2022, 2022, pp. 116–121. doi: 10.1109/TSP55681.2022.9851258.

[74] C. Gong, W. He, T. Wang, A. Gani, and H. Qi, “Dynamic resource allocation scheme for mobile edge computing,” Journal of Supercomputing, vol. 79, no. 15, pp. 17187–17207, 2023, doi: 10.1007/s11227-023-05323-y.

[75] Z. Ali et al., “Edge Based Priority-Aware Dynamic Resource Allocation for Internet of Things Networks,” Entropy, vol. 24, no. 11, 2022, doi: 10.3390/e24111607.

[76] M. Chochul and P. Sevcik, “A Survey of Low Power Wide Area Network Technologies,” in ICETA 2020 - 18th IEEE International Conference on Emerging eLearning Technologies and Applications, Proceedings, 2020, pp. 69–73. doi: 10.1109/ICETA51985.2020.9379213.

[77] Y. Mansouri, V. Prokhorenko, F. Ullah, and M. A. Babar, “Resource Utilization of Distributed Databases in Edge-Cloud Environment,” IEEE Internet Things J, vol. 10, no. 11, pp. 9423–9437, 2023, doi: 10.1109/JIOT.2023.3235360.

[78] M. M. Kadum and X. Deng, “Allocating energy-objective aware workflow in distributed edge micro data centres,” Journal of Supercomputing, 2023, doi: 10.1007/s11227-023-05681-7.

[79] K. M. P. Maurya and A. A. Kherani, “A Modular Hybrid Simulator for LoRaWAN,” in International Symposium on Advanced Networks and Telecommunication Systems, ANTS, 2020. doi: 10.1109/ANTS50601.2020.9342777.

[80] L. Zhu, Z. Zhang, L. Liu, L. Feng, P. Lin, and Y. Zhang, “Online Distributed Learning-Based Load-Aware Heterogeneous Vehicular Edge Computing,” IEEE Sens J, vol. 23, no. 15, pp. 17350–17365, 2023, doi: 10.1109/JSEN.2023.3283413.

[81] A. Durand, P. Gremaud, and J. Pasquier, “Decentralized LPWAN infrastructure using blockchain and digital signatures,” Concurr Comput, vol. 32, no. 12, 2020, doi: 10.1002/cpe.5352.

[82] N. A. Abdul Latiff, I. S. Ismail, M. H. Yusoff, A. R. Salisa, and J. A. Shukor, Scalability Analysis of Low-Power Wide Area Network Technology, vol. 1183. 2021. doi: 10.1007/978-981-15-5856-6_12.

[83] M. I. Hossain and J. I. Markendahl, “Comparison of LPWAN Technologies: Cost Structure and Scalability,” Wirel Pers Commun, vol. 121, no. 1, pp. 887–903, 2021, doi: 10.1007/s11277-021-08664-0.

[84] V. Kalokidou, M. Nair, and M. A. Beach, “LoRaWAN Performance Evaluation and Resilience under Jamming Attacks,” in 2022 Sensor Signal Processing for Defence Conference, SSPD 2022 - Proceedings, 2022. doi: 10.1109/SSPD54131.2022.9896225.

[85] M. Bolourian and H. Shah-Mansouri, “Energy-Efficient Task Offloading for Three-Tier Wireless-Powered Mobile-Edge Computing,” IEEE Internet Things J, vol. 10, no. 12, pp. 10400–10412, 2023, doi: 10.1109/JIOT.2023.3238329.

[86] D. Kimovski, R. Matha, J. Hammer, N. Mehran, H. Hellwagner, and R. Prodan, “Cloud, Fog, or Edge: Where to Compute?,” IEEE Internet Comput, vol. 25, no. 4, pp. 30–36, 2021, doi: 10.1109/MIC.2021.3050613.

[87] K. Peng, H. Huang, P. Liu, X. Xu, and V. C. M. Leung, “Joint Optimization of Energy Conservation and Privacy Preservation for Intelligent Task Offloading in MEC-Enabled Smart Cities,” IEEE Transactions on Green Communications and Networking, vol. 6, no. 3, pp. 1671–1682, 2022, doi: 10.1109/TGCN.2022.3170146.

[88] A. Lekidis and E. I. Papageorgiou, “Edge-Based Short-Term Energy Demand Prediction,” Energies (Basel), vol. 16, no. 14, 2023, doi: 10.3390/en16145435.

[89] Z. Ning et al., “Mobile Edge Computing and Machine Learning in the Internet of Unmanned Aerial Vehicles: A Survey,” ACM Comput Surv, vol. 56, no. 1, 2023, doi: 10.1145/3604933.

[90] B. Aiswarya Padma, “Connected vehicles using LoRa,” in 2020 7th International Conference on Smart Structures and Systems, ICSSS 2020, 2020. doi: 10.1109/ICSSS49621.2020.9202133.

[91] M. K. Sharma, R. Mehta, and R. S. Shekhawat, IoT enabled by edge computing for telecomms and industry. 2023.

[92] G. Czeczot, I. Rojek, D. Mikołajewski, and B. Sangho, “AI in IIoT Management of Cybersecurity for Industry 4.0 and Industry 5.0 Purposes,” Electronics (Switzerland), vol. 12, no. 18, 2023, doi: 10.3390/electronics12183800.

[93] F. Battaglia, G. Gugliandolo, G. Campobello, and N. Donato, “Monitoring sea pollution using wireless QCM-based sensors,” in 2022 IEEE International Workshop on Metrology for the Sea; Learning to Measure Sea Health Parameters, MetroSea 2022 - Proceedings, 2022, pp. 462–467. doi: 10.1109/MetroSea55331.2022.9950953.

[94] P. Jain and P. Chawla, “A Novel Smart Healthcare System Design for Internet of Health Things,” in Proceedings of the 2021 IEEE International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems, ICSES 2021, 2021. doi: 10.1109/ICSES52305.2021.9633869.

[95] S. Tuli et al., “AI augmented Edge and Fog computing: Trends and challenges,” Journal of Network and Computer Applications, vol. 216, 2023, doi: 10.1016/j.jnca.2023.103648.

[96] E. Psomakelis, A. Makris, K. Tserpes, and M. Pateraki, “A lightweight storage framework for edge computing infrastructures/EdgePersist,” Software Impacts, vol. 17, 2023, doi: 10.1016/j.simpa.2023.100549.

[97] A. S. Shaker, “A Survey of Smart Buildings and Homes using Low-Power Wide-Area Network (LoRa WAN),” in 4th International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT 2020 - Proceedings, 2020. doi: 10.1109/ISMSIT50672.2020.9254675.

[98] A. Mesodiakaki et al., “5G-COMPLETE: End-to-end Resource Allocation in Highly Heterogeneous beyond 5G Networks,” in Proceedings - 2021 IEEE 4th 5G World Forum, 5GWF 2021, 2021, pp. 412–417. doi: 10.1109/5GWF52925.2021.00079.

[99] H. Zhang, C. Li, Z. Hua, W. Ren, Y. Gao, and Z. Hao, “Design of aquaculture monitoring system based on LoRa and 4G network of internet of things,” in Proceedings of SPIE - The International Society for Optical Engineering, 2021. doi: 10.1117/12.2601041.

[100] F. Zhou, L. Feng, M. Kadoch, P. Yu, W. Li, and Z. Wang, “Multiagent RL Aided Task Offloading and Resource Management in Wi-Fi 6 and 5G Coexisting Industrial Wireless Environment,” IEEE Trans Industr Inform, vol. 18, no. 5, pp. 2923–2933, 2022, doi: 10.1109/TII.2021.3106973.

[101] F. Zhao, X. Wang, P. Lin, and Y. Chen, “Comprehensive analysis of the heterogeneous computing performance of DNNs under typical frameworks on cloud and edge computing platforms,” Expert Syst Appl, vol. 229, 2023, doi: 10.1016/j.eswa.2023.120475.

[102] W. Fu, Y. Wan, J. Qin, Y. Kang, and L. Li, “Privacy-Preserving Optimal Energy Management for Smart Grid With Cloud-Edge Computing,” IEEE Trans Industr Inform, vol. 18, no. 6, pp. 4029–4038, 2022, doi: 10.1109/TII.2021.3114513.

[103] Y. Zhiqiang, X. Xin, R. Hui, W. Yanpin, and R. Hao, “Research on service-oriented communication technology supporting inter-regional information security interaction in substation,” in 2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), IEEE, May 2023, pp. 342–348. doi: 10.1109/ICIBA56860.2023.10165191.

[104] M. M. Adam, L. Zhao, K. Wang, and Z. Han, “Beyond 5G networks: Integration of communication, computing, caching, and control,” China Communications, vol. 20, no. 7, pp. 137–174, 2023, doi: 10.23919/JCC.2023.00.039.

[105] M. Oproiu, A. Neagu, P. A. Cotfas, D. T. Cotfas, C. Musuroi, and M. Volmer, “LoRa Wide-Area Network and Live Objects Used in Renewable Energy Monitoring,” in 2021 International Aegean Conference on Electrical Machines and Power Electronics, ACEMP 2021 and 2021 International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2021, 2021, pp. 505–512. doi: 10.1109/OPTIM-ACEMP50812.2021.9590023.

[106] C. Guleria, K. Das, and A. Sahu, “A Survey on Mobile Edge Computing: Efficient Energy Management System,” in 2021 Innovations in Energy Management and Renewable Resources, IEMRE 2021, 2021. doi: 10.1109/IEMRE52042.2021.9386951.

[107] Q. Vu Khanh, N. Vi Hoai, A. Dang Van, and Q. Nguyen Minh, “An integrating computing framework based on edge-fog-cloud for internet of healthcare things applications,” Internet of Things, vol. 23, p. 100907, Oct. 2023, doi: 10.1016/j.iot.2023.100907.

[108] C. Chakraborty, S. M. Nagarajan, G. G. Devarajan, T. V Ramana, and R. Mohanty, “Intelligent AI-based Healthcare Cyber Security System using Multi-Source Transfer Learning Method,” ACM Trans Sens Netw, May 2023, doi: 10.1145/3597210.

[109] A. Singh and K. Chatterjee, “Edge computing based secure health monitoring framework for electronic healthcare system,” Cluster Comput, vol. 26, no. 2, pp. 1205–1220, Apr. 2023, doi: 10.1007/s10586-022-03717-w.

[110] A. Upadhyaya and M. K. Mishra, “Authentication Schemes for Healthcare Data Using Emerging Computing Technologies,” 2023, pp. 67–82. doi: 10.1007/978-981-99-0377-1_5.

[111] Y. Wu, Y. He, and L. Shi, “Energy-saving measurement in LoRaWAN-Based wireless sensor networks by using compressed sensing,” IEEE Access, vol. 8, pp. 49477–49486, 2020, doi: 10.1109/ACCESS.2020.2974879.

[112] D. Li, M. Lan, and Y. Hu, “Energy-saving service management technology of internet of things using edge computing and deep learning,” Complex and Intelligent Systems, 2022, doi: 10.1007/s40747-022-00666-0.

[113] Q. Zhang, X. Lin, Y. Hao, and J. Cao, “Energy-Aware Scheduling in Edge Computing Based on Energy Internet,” IEEE Access, vol. 8, pp. 229052–229065, 2020, doi: 10.1109/ACCESS.2020.3044932.

[114] S. S. Mahadik, P. M. Pawar, and R. Muthalagu, “Edge-HetIoT defense against DDoS attack using learning techniques,” Comput Secur, vol. 132, 2023, doi: 10.1016/j.cose.2023.103347.

[115] Z. Xiang, S. Deng, Y. Zheng, D. Wang, J. Tehari, and Z. Zheng, “Energy-effective IoT Services in Balanced Edge-Cloud Collaboration Systems,” in Proceedings - 2021 IEEE International Conference on Web Services, ICWS 2021, 2021, pp. 219–229. doi: 10.1109/ICWS53863.2021.00040.

[116] K. Gai et al., “Attacking the Edge-of-Things: A Physical Attack Perspective,” IEEE Internet Things J, vol. 9, no. 7, pp. 5240–5253, Apr. 2022, doi: 10.1109/JIOT.2021.3109917.

[117] Y. K. Saheed, “Performance Improvement of Intrusion Detection System for Detecting Attacks on Internet of Things and Edge of Things,” 2022, pp. 321–339. doi: 10.1007/978-3-030-80821-1_15.

[118] T. P. Truong, D. T. Nguyen, and P. V. Truong, “Design and deployment of an IoT-based air quality monitoring system,” International Journal of Environmental Science and Development, vol. 12, no. 3, pp. 139–145, 2021, doi: 10.18178/IJESD.2021.12.5.1331.

[119] Q. Sui and X. Liu, “Edge computing and AIoT based network intrusion detection mechanism,” Internet Technology Letters, vol. 6, no. 5, 2023, doi: 10.1002/itl2.324.

[120] R. Yang et al., “Efficient intrusion detection toward IoT networks using cloud–edge collaboration,” Computer Networks, vol. 228, 2023, doi: 10.1016/j.comnet.2023.109724.

[121] B. C. Mallikarjun, K. J. Kiranmayi, N. Lavanya, K. H. Prateeksha, and J. Sushmitha, “Intruder detection system - A lora based approach,” in Proceedings of the 5th International Conference on Communication and Electronics Systems, ICCES 2020, 2020, pp. 255–258. doi: 10.1109/ICCES48766.2020.09137923.

[122] A. Khalifeh, K. Aldahdouh, and S. Alouneh, “Optimizing the energy consumption level in LoraWan networks,” in Proceedings - 2020 21st International Arab Conference on Information Technology, ACIT 2020, 2020. doi: 10.1109/ACIT50332.2020.9300117.

[123] A. Mohamed et al., “Enhancing Cyber Security of LoRaWAN Gateways under Adversarial Attacks,” Sensors, vol. 22, no. 9, 2022, doi: 10.3390/s22093498.

[124] H. Gami, “Cost-effective Energy Optimization and Indoor Surveillance,” in IEACon 2021 - 2021 IEEE Industrial Electronics and Applications Conference, 2021, pp. 292–296. doi: 10.1109/IEACon51066.2021.9654683.

[125] S. Ahmad et al., “Sustainable Environmental Monitoring via Energy and Information Efficient Multi-Node Placement,” IEEE Internet Things J, 2023, doi: 10.1109/JIOT.2023.3303124.

[126] M.-Y. Wu, C.-J. Tsai, W.-T. Su, S. Ciou, Y.-S. Lee, and C.-H. Lee, “Hybrid Cloud Energy Management for Edge Computing,” in 2023 International Conference on Consumer Electronics - Taiwan, ICCE-Taiwan 2023 - Proceedings, 2023, pp. 143–144. doi: 10.1109/ICCE-Taiwan58799.2023.10226741.

[127] F. Cicirelli, A. F. Gentile, E. Greco, A. Guerrieri, G. Spezzano, and A. Vinci, “An Energy Management System at the Edge based on Reinforcement Learning,” in Proceedings of the 2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications, DS-RT 2020, 2020. doi: 10.1109/DS-RT50469.2020.9213697.

[128] T. Szydlo and C. Gniady, Renewable Energy-Aware Heuristic Algorithms for Edge Server Selection for Stream Data Processing, vol. 12745 LNCS. 2021. doi: 10.1007/978-3-030-77970-2_49.

[129] M. Avgeris, D. Spatharakis, D. Dechouniotis, A. Leivadeas, V. Karyotis, and S. Papavassiliou, “ENERDGE: Distributed Energy-Aware Resource Allocation at the Edge,” Sensors, vol. 22, no. 2, 2022, doi: 10.3390/s22020660.

[130] S. Popli, R. K. Jha, and S. Jain, “Adaptive Small Cell position algorithm (ASPA) for green farming using NB-IoT,” Journal of Network and Computer Applications, vol. 173, p. 102841, Jan. 2021, doi: 10.1016/j.jnca.2020.102841.

[131] T. Singh, A. Solanki, and S. K. Sharma, "Predictive Analysis of Energy Consumption for Energy Management in Smart Homes," 2023, pp. 587–600. doi: 10.1007/978-981-19-9304-6_53.

[132] Y. Cao, A. Wang, G. Sun, and L. Liu, "Average Transmission Rate and Energy Efficiency Optimization in UAV-assisted IoT," in 2023 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, Mar. 2023, pp. 1–6. doi: 10.1109/WCNC55385.2023.10119068.

[133] H. Makina and A. Ben Letaifa, "Bringing intelligence to Edge/Fog in Internet of Things-based healthcare applications: Machine learning/deep learning-based use cases," International Journal of Communication Systems, vol. 36, no. 9, 2023, doi: 10.1002/dac.5484.

[134] M. Gonzalez-Palacio, D. Tobon-Vallejo, L. M. Sepulveda-Cano, S. Rua, and L. B. Le, "Machine-Learning-Based Combined Path Loss and Shadowing Model in LoRaWAN for Energy Efficiency Enhancement," IEEE Internet Things J, vol. 10, no. 12, pp. 10725–10739, 2023, doi: 10.1109/JIOT.2023.3239827.

[135] V. K. Kaliappan et al., "Energy-Efficient Offloading Based on Efficient Cognitive Energy Management Scheme in Edge Computing Device with Energy Optimization," Energies (Basel), vol. 15, no. 21, 2022, doi: 10.3390/en15218273.

[136] J. A. Alzubi, O. A. Alzubi, A. Singh, and T. Mahmod Alzubi, "A blockchain-enabled security management framework for mobile edge computing," International Journal of Network Management, vol. 33, no. 5, 2023, doi: 10.1002/nem.2240.

[137] L. Yang, M. Li, Y. Zhang, P. Si, Z. Wang, and R. Yang, "Resource Management for Energy-Efficient and Blockchain-Enabled Industrial IoT: A DRL Approach," in 2020 IEEE 6th International Conference on Computer and Communications, ICCC 2020, 2020, pp. 910–915. doi: 10.1109/ICCC51575.2020.9345166.

[138] S. Abed, R. Jaffal, and B. J. Mohd, "A Review on Blockchain and IoT Integration from Energy, Security and Hardware Perspectives," Wirel Pers Commun, vol. 129, no. 3, pp. 2079–2122, 2023, doi: 10.1007/s11277-023-10226-5.

[139] T. R. Gadekallu et al., "Blockchain for Edge of Things: Applications, Opportunities, and Challenges," IEEE Internet Things J, vol. 9, no. 2, pp. 964–988, Jan. 2022, doi: 10.1109/JIOT.2021.3119639.

[140] A. Akhunzada, S. Zeadally, and S. ul Islam, "Toward Performance and Energy-Efficient Edge-of-Things," IT Prof, vol. 25, no. 1, pp. 17–23, Jan. 2023, doi: 10.1109/MITP.2022.3230837.

[141] X. Ma, A. Zhou, S. Zhang, Q. Li, A. X. Liu, and S. Wang, "Dynamic Task Scheduling in Cloud-Assisted Mobile Edge Computing," IEEE Trans Mob Comput, vol. 22, no. 4, pp. 2116–2130, 2023, doi: 10.1109/TMC.2021.3115262.

[142] M. T. Quasim, "Resource Management and Task Scheduling for IoT using Mobile Edge Computing," Wirel Pers Commun, 2021, doi: 10.1007/s11277-021-09087-7.

[143] J. Feng, W. Zhang, L. Liu, J. Du, M. Xiao, and Q. Pei, Task Scheduling and Resource Management in MEC-Enabled Computing Networks, vol. 418 LNICST. 2022. doi: 10.1007/978-3-030-94763-7_10.

[144] M.-W. Tian, S.-R. Yan, W. Guo, A. Mohammadzadeh, and E. Ghaderpour, “A New Task Scheduling Approach for Energy Conservation in Internet of Things,” Energies (Basel), vol. 16, no. 5, 2023, doi: 10.3390/en16052394.

[145] E. Al-Masri, A. Souri, H. Mohamed, W. Yang, J. Olmsted, and O. Kotevska, “Energy-efficient cooperative resource allocation and task scheduling for Internet of Things environments,” Internet of Things (Netherlands), vol. 23, 2023, doi: 10.1016/j.iot.2023.100832.

[146] F. Marozzo, A. Orsino, D. Talia, and P. Trunfio, “Edge Computing Solutions for Distributed Machine Learning,” in Proceedings of the 2022 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cy, 2022. doi: 10.1109/DASC/PiCom/CBDCom/Cy55231.2022.9927824.

[147] Z. Zhong, J. He, M. A. Rodriguez, S. Erfani, R. Kotagiri, and R. Buyya, “Heterogeneous Task Co-location in Containerized Cloud Computing Environments,” in 2020 IEEE 23rd International Symposium on Real-Time Distributed Computing (ISORC), IEEE, May 2020, pp. 79–88. doi: 10.1109/ISORC49007.2020.00021.

[148] S. Khatua, D. Manerba, S. Maity, and D. De, “Dew Computing-Based Sustainable Internet of Vehicular Things,” 2024, pp. 181–205. doi: 10.1007/978-981-99-4590-0_9.

[149] K. Le Nguyen, H. Thi Trinh, T. T. Nguyen, and H. D. Nguyen, “Comparative study on the performance of different machine learning techniques to predict the shear strength of RC deep beams: Model selection and industry implications,” Expert Syst Appl, vol. 230, p. 120649, Nov. 2023, doi: 10.1016/j.eswa.2023.120649.

[150] H. Xu, Z. Sun, Y. Cao, and H. Bilal, “A data-driven approach for intrusion and anomaly detection using automated machine learning for the Internet of Things,” Soft comput, vol. 27, no. 19, pp. 14469–14481, Oct. 2023, doi: 10.1007/s00500-023-09037-4.

[151] T. Zhou, D. Qin, X. Nie, X. Li, and C. Li, “Energy-efficient computation offloading and resource management in ultradense heterogeneous networks,” IEEE Trans Veh Technol, vol. 70, no. 12, pp. 13101–13114, 2021, doi: 10.1109/TVT.2021.3116955.

[152] G. Cui, Q. He, X. Xia, F. Chen, and Y. Yang, “Energy-efficient Edge Server Management for Edge Computing: A Game-theoretical Approach,” in ACM International Conference Proceeding Series, 2022. doi: 10.1145/3545008.3545079.

[153] F. Li and C. Wang, “Artificial intelligence and edge computing for teaching quality evaluation based on 5G-enabled wireless communication technology,” Journal of Cloud Computing, vol. 12, no. 1, 2023, doi: 10.1186/s13677-023-00418-6.

[154] V. Patsias, P. Amanatidis, D. Karampatzakis, T. Lagkas, K. Michalakopoulou, and A. Nikitas, “Task Allocation Methods and Optimization Techniques in Edge Computing: A

Systematic Review of the Literature," Future Internet, vol. 15, no. 8, 2023, doi: 10.3390/fi15080254.

[155] E. Fathy, E. Ammar, and M. A. Helmy, "Three different optimization techniques for solving the fully rough interval multi-level linear programming problem," Journal of Intelligent and Fuzzy Systems, vol. 45, no. 2, pp. 2593–2610, 2023, doi: 10.3233/JIFS-230057.

[156] S. A. Rocha, T. G. Mattos, R. T. N. Cardoso, and E. G. Silveira, "Applying Artificial Neural Networks and Nonlinear Optimization Techniques to Fault Location in Transmission Lines—Statistical Analysis," Energies (Basel), vol. 15, no. 11, 2022, doi: 10.3390/en15114095.

[157] A. V. H. Sai Prasad and G. V. S. Rajkumar, "A Novel Dynamic Optimization Technique for Finding Optimal Trust Weights in Cloud," KSII Transactions on Internet and Information Systems, vol. 16, no. 6, pp. 2060–2073, 2022, doi: 10.3837/tiis.2022.06.015.

[158] L. Cai, K. Yao, W. Xian, and C. Gong, "Edge Computing Resource Management Based on Genetic Algorithm," in ICEIEC 2023 - Proceedings of 2023 IEEE 13th International Conference on Electronics Information and Emergency Communication, 2023, pp. 153–157. doi: 10.1109/ICEIEC58029.2023.10199776.

[159] J. Zhang et al., Low-Energy Edge Computing Resource Deployment Algorithm Based on Particle Swarme, vol. 1274 AISC. 2021. doi: 10.1007/978-981-15-8462-6_178.

[160] J. Liu, P. Yang, and C. Chen, "Intelligent energy-efficient scheduling with ant colony techniques for heterogeneous edge computing," J Parallel Distrib Comput, vol. 172, pp. 84–96, 2023, doi: 10.1016/j.jpdc.2022.10.003.

[161] S. K. Sunori, P. K. Juneja, G. S. Jethi, A. Bhakuni, and M. Chaturvedi, Controller optimization for boiler turbine using simulated annealing and genetic algorithm, vol. 1086. 2021. doi: 10.1007/978-981-15-1275-9_4.

[162] H. Li, S. Han, X. Wu, and F. Wang, "A Novel Task of Loading and Computing Resource Scheduling Strategy in Internet of Vehicles Based on Dynamic Greedy Algorithm," Tehnicki Vjesnik, vol. 30, no. 4, pp. 1298–1307, 2023, doi: 10.17559/TV-20221207032927.

[163] M. Rabbani, M. Mokhtarzadeh, and N. Manavizadeh, "A constraint programming approach and a hybrid of genetic and K-means algorithms to solve the p -hub location-allocation problems," International Journal of Management Science and Engineering Management, vol. 16, no. 2, pp. 123–133, Apr. 2021, doi: 10.1080/17509653.2021.1905096.

[164] S. Taheri-abed, A. M. Eftekhari Moghadam, and M. H. Rezvani, "Machine learning-based computation offloading in edge and fog: a systematic review," Cluster Comput, vol. 26, no. 5, pp. 3113–3144, 2023, doi: 10.1007/s10586-023-04100-z.

[165] Y. Gong, K. Bian, F. Hao, Y. Sun, and Y. Wu, "Dependent tasks offloading in mobile edge computing: A multi-objective evolutionary optimization strategy," Future Generation Computer Systems, vol. 148, pp. 314–325, 2023, doi: 10.1016/j.future.2023.06.015.

[166] D. Hortelano et al., “A comprehensive survey on reinforcement-learning-based computation offloading techniques in Edge Computing Systems,” Journal of Network and Computer Applications, vol. 216, 2023, doi: 10.1016/j.jnca.2023.103669.

[167] H. Badri, T. Bahreini, D. Grosu, and K. Yang, “Energy-Aware application placement in mobile edge computing: A stochastic optimization approach,” IEEE Transactions on Parallel and Distributed Systems, vol. 31, no. 4, pp. 909–922, 2020, doi: 10.1109/TPDS.2019.2950937.

[168] N. Alhebaishi et al., “Soft Computing Based Metaheuristic Algorithms for Resource Management in Edge Computing Environment,” Computers, Materials and Continua, vol. 72, no. 3, pp. 5233–5250, 2022, doi: 10.32604/cmc.2022.025596.

[169] H. Feng, L. Qiao, and Z. Lv, “Innovative soft computing-enabled cloud optimization for next-generation IoT in digital twins,” Appl Soft Comput, vol. 136, 2023, doi: 10.1016/j.asoc.2023.110082.

[170] L. Yang, Y. Zou, D. Yu, and J. Yu, “Distributed Age-of-Information optimization in edge computing for Internet of Vehicles,” Journal of Systems Architecture, vol. 144, 2023, doi: 10.1016/j.sysarc.2023.103000.

[171] C. Lehong, B. Isong, F. Lugayizi, and A. M. Abu-Mahfouz, “A Survey of LoRaWAN Adaptive Data Rate Algorithms for Possible Optimization,” in 2020 2nd International Multidisciplinary Information Technology and Engineering Conference, IMITEC 2020, 2020. doi: 10.1109/IMITEC50163.2020.9334144.

[172] F. Cuomo, D. Garlisi, A. Martino, and A. Martino, “Predicting lorawan behavior: How machine learning can help,” Computers, vol. 9, no. 3, pp. 1–18, 2020, doi: 10.3390/computers9030060.

[173] L. Liu, “A Bayesian Deep Learning Network System Based on Edge Computing,” International Journal of Humanoid Robotics, vol. 20, no. 2–3, 2023, doi: 10.1142/S0219843622500086.

[174] C. Gao, Y. Chen, Y. Chen, Z. Wang, and H. Xia, “An improved k-NN anomaly detection framework based on locality sensitive hashing for edge computing environment,” Intelligent Data Analysis, vol. 27, no. 5, pp. 1267–1285, 2023, doi: 10.3233/IDA-216461.

[175] V. D. Pham, P. Hao Do, D. T. Le, and R. Kirichek, LoRa Link Quality Estimation Based on Support Vector Machine, vol. 13144 LNCS. 2021. doi: 10.1007/978-3-030-92507-9_9.

[176] G. Cassales, H. M. Gomes, A. Bifet, B. Pfahringer, and H. Senger, “Balancing Performance and Energy Consumption of Bagging Ensembles for the Classification of Data Streams in Edge Computing,” IEEE Transactions on Network and Service Management, vol. 20, no. 3, pp. 3038–3054, 2023, doi: 10.1109/TNSM.2022.3226505.

[177] W. Ingabire, H. Larijani, R. M. Gibson, and A.-U.-H. Qureshi, “LoRaWAN Based Indoor Localization Using Random Neural Networks,” Information (Switzerland), vol. 13, no. 6, 2022, doi: 10.3390/info13060303.

[178] A. Lazaro, M. Lazaro, and R. Villarino, “Room-Level Localization System Based on LoRa Backscatters,” IEEE Access, vol. 9, pp. 16004–16018, 2021, doi: 10.1109/ACCESS.2021.3053144.

[179] Y. Wang et al., “A Federated Network Intrusion Detection System with Multi-Branch Network and Vertical Blocking Aggregation,” Electronics (Switzerland), vol. 12, no. 19, 2023, doi: 10.3390/electronics12194049.

[180] T. Le-Anh, Q. Ngo-Van, P. Vo-Huy, D. Huynh-Van, and Q. Le-Trung, A Container-Based Edge Computing System for Smart Healthcare Applications, vol. 379. 2021. doi: 10.1007/978-3-030-77424-0_27.

[181] Y. Yazid, I. Ez-Zazi, M. Arioua, and A. E. L. Oualkadi, “On the LoRa performances under different physical layer parameter selection,” in 2020 International Symposium on Advanced Electrical and Communication Technologies, ISAECT 2020, 2020. doi: 10.1109/ISAECT50560.2020.9523690.

[182] I. S. Ismail, N. A. Abdul Latiff, N. A. Azmi Ali, and N. M. Abdul Latiff, “Performance evaluation of physical layer using lora technology for indoor environment,” International Journal of Scientific and Technology Research, vol. 9, no. 1, pp. 36–41, 2020.

[183] W. J. Hou, Y. Jiang, W. Lei, A. Xu, H. Wen, and S. Chen, “A P2P network based edge computing smart grid model for efficient resources coordination,” Peer Peer Netw Appl, vol. 13, no. 3, pp. 1026–1037, 2020, doi: 10.1007/s12083-019-00870-9.

[184] Z. Tang, J. Lou, and W. Jia, “Layer Dependency-Aware Learning Scheduling Algorithms for Containers in Mobile Edge Computing,” IEEE Trans Mob Comput, vol. 22, no. 6, pp. 3444–3459, 2023, doi: 10.1109/TMC.2021.3139995.

[185] H. Taleb, A. Nasser, G. Andrieux, N. Charara, and E. M. Cruz, “Energy Consumption Improvement of a Healthcare Monitoring System: Application to LoRaWAN,” IEEE Sens J, vol. 22, no. 7, pp. 7288–7299, 2022, doi: 10.1109/JSEN.2022.3150716.

[186] G. Callebaut, G. Ottoy, and L. V. D. Perre, “Optimizing Transmission of IoT Nodes in Dynamic Environments,” in 2020 International Conference on Omni-Layer Intelligent Systems, COINS 2020, 2020. doi: 10.1109/COINS49042.2020.9191674.

[187] V. F. Rodrigues et al., “Digital health in smart cities: Rethinking the remote health monitoring architecture on combining edge, fog, and cloud,” Health Technol (Berl), vol. 13, no. 3, pp. 449–472, 2023, doi: 10.1007/s12553-023-00753-3.

[188] S. Stoykova and N. Shakev, “Artificial Intelligence for Management Information Systems: Opportunities, Challenges, and Future Directions,” Algorithms, vol. 16, no. 8, 2023, doi: 10.3390/a16080357.

[189] T. Wang, Y. Liang, X. Shen, X. Zheng, A. Mahmood, and Q. Z. Sheng, “Edge Computing and Sensor-Cloud: Overview, Solutions, and Directions,” ACM Comput Surv, vol. 55, no. 13s, 2023, doi: 10.1145/3582270.

[190] M. Ballerini, T. Polonelli, D. Brunelli, M. Magno, and L. Benini, "NB-IoT Versus LoRaWAN: An Experimental Evaluation for Industrial Applications," IEEE Trans Industr Inform, vol. 16, no. 12, pp. 7802–7811, 2020, doi: 10.1109/TII.2020.2987423.

[191] T. Brito et al., "Data Acquisition Filtering Focused on Optimizing Transmission in a LoRaWAN Network Applied to the WSN Forest Monitoring System," Sensors, vol. 23, no. 3, 2023, doi: 10.3390/s23031282.

[192] C. Wang et al., "Parallel in-memory wireless computing," Nat Electron, vol. 6, no. 5, pp. 381–389, 2023, doi: 10.1038/s41928-023-00965-5.

[193] Md. Al Maruf and A. Azim, "A software-based calibration approach to increase the robustness of embedded systems," Int J Embed Syst, vol. 13, no. 4, pp. 466–475, 2020, doi: 10.1504/IJES.2020.110662.

[194] S. Ozer, H. E. Ilhan, M. A. Ozkanoglu, and H. A. Cirpan, "Offloading Deep Learning Powered Vision Tasks from UAV to 5G Edge Server with Denoising," IEEE Trans Veh Technol, vol. 72, no. 6, pp. 8035–8048, 2023, doi: 10.1109/TVT.2023.3243529.

[195] G. Guo, "Real-time medical image denoising and information hiding model based on deep wavelet multiscale autonomous unmanned analysis," Soft comput, vol. 27, no. 7, pp. 4263–4278, 2023, doi: 10.1007/s00500-022-07322-2.

[196] A. Singh and K. Chatterjee, "Edge computing based secure health monitoring framework for electronic healthcare system," Cluster Comput, vol. 26, no. 2, pp. 1205–1220, 2023, doi: 10.1007/s10586-022-03717-w.

[197] S. Ahmed, S. Irfan, N. Kiran, N. Masood, N. Anjum, and N. Ramzan, "Remote Health Monitoring Systems for Elderly People: A Survey," Sensors, vol. 23, no. 16, 2023, doi: 10.3390/s23167095.

[198] A. Valach and D. MacKo, "Optimization of LoRa Devices Communication for Applications in Healthcare," in 2020 43rd International Conference on Telecommunications and Signal Processing, TSP 2020, 2020, pp. 511–514. doi: 10.1109/TSP49548.2020.9163432.

[199] S. Ahmed, S. Irfan, N. Kiran, N. Masood, N. Anjum, and N. Ramzan, "Remote Health Monitoring Systems for Elderly People: A Survey," Sensors, vol. 23, no. 16, 2023, doi: 10.3390/s23167095.

[200] S. Elango, K. P. Sampoornam, S. Ilakkiya, N. Harshitha, and S. Janani, LoRaWAN-Based Intelligent Home and Health Monitoring of Elderly People, vol. 995 LNEE. 2023. doi: 10.1007/978-981-19-9748-8_22.

[201] J. P. Lousado and S. Antunes, "E-Health Monitoring System for Senior Citizens based on LoRa Technology," in 2020 5th International Conference on Smart and Sustainable Technologies, SpliTech 2020, 2020. doi: 10.23919/SpliTech49282.2020.9243835.

[202] R. Balaji V, R. Dinesh Kumar J, A. Arivarasan, J. K. Raja, and P. Chidambaram, "Internet of Things based Smart Health Care System using LoRa," in Proceedings - 2nd

International Conference on Smart Electronics and Communication, ICOSEC 2021, 2021, pp. 89–95. doi: 10.1109/ICOSEC51865.2021.9591851.

[203] Z.-H. Wang, S.-T. Shih, H. Hendrick, M.-Y. Pai, and G.-J. Horng, “Deployment and Evaluation of LoRa Network Configuration Based on Random Forest,” in Proceedings - 2020 International Computer Symposium, ICS 2020, 2020, pp. 262–265. doi: 10.1109/ICS51289.2020.00059.

[204] S. A. Magsi, M. H. B. M. Khir, I. B. M. Nawi, A. Saboor, and M. A. Siddiqui, “Support Vector Regression based Localization Approach using LoRaWAN,” International Journal of Advanced Computer Science and Applications, vol. 14, no. 3, pp. 307–312, 2023, doi: 10.14569/IJACSA.2023.0140335.

[205] T. P. Dinh, C. Pham-Quoc, T. N. Thinh, B. K. D. Nguyen, and P. C. Kha, “A flexible and efficient FPGA-based random forest architecture for IoT applications,” Internet of Things (Netherlands), vol. 22, 2023, doi: 10.1016/j.iot.2023.100813.

[206] S. Gupta and I. Snigdh, “Clustering in LoRa Networks, an Energy-Conserving Perspective,” Wirel Pers Commun, vol. 122, no. 1, pp. 197–210, 2022, doi: 10.1007/s11277-021-08894-2.

[207] Y. Li et al., “Energy-Aware, Device-to-Device Assisted Federated Learning in Edge Computing,” IEEE Transactions on Parallel and Distributed Systems, 2023, doi: 10.1109/TPDS.2023.3277423.

[208] Y. Lykov, A. Paniotova, V. Shatalova, and A. Lykova, “Energy Efficiency Comparison LPWANs: LoRaWAN vs Sigfox,” in 2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020 - Proceedings, 2021, pp. 485–490. doi: 10.1109/PICST51311.2020.9468026.

[209] M. Perez et al., “Coverage and Energy-Efficiency Experimental Test Performance for a Comparative Evaluation of Unlicensed LPWAN: LoRaWAN and SigFox,” IEEE Access, vol. 10, pp. 97183–97196, 2022, doi: 10.1109/ACCESS.2022.3206030.

[210] A. Loubany, S. Lahoud, A. E. Samhat, and M. El Helou, “Improving Energy Efficiency in LoRaWAN Networks with Multiple Gateways,” Sensors, vol. 23, no. 11, 2023, doi: 10.3390/s23115315.

[211] G. Zhao, K. Lin, D. Chapman, N. Metje, and T. Hao, “Optimizing energy efficiency of LoRaWAN-based wireless underground sensor networks: A multi-agent reinforcement learning approach,” Internet of Things (Netherlands), vol. 22, 2023, doi: 10.1016/j.iot.2023.100776.

[212] R. K. Singh, R. Berkvens, and M. Weyn, “Synchronization and efficient channel hopping for power efficiency in LoRa networks: A comprehensive study,” Internet of Things (Netherlands), vol. 11, 2020, doi: 10.1016/j.iot.2020.100233.

[213] K. Djemame, Energy Efficiency in Edge Environments: a Serverless Computing Approach, vol. 13072 LNCS. 2021. doi: 10.1007/978-3-030-92916-9_15.

[214] J. P. Lousado and S. Antunes, “Monitoring and support for elderly people using lora communication technologies: Iot concepts and applications,” Future Internet, vol. 12, no. 11, pp. 1–30, 2020, doi: 10.3390/fi12110206.

[215] J. P. Lousado, I. M. Pires, E. Zdravevski, and S. Antunes, “Monitoring the health and residence conditions of elderly people, using lora and the things network,” Electronics (Switzerland), vol. 10, no. 14, 2021, doi: 10.3390/electronics10141729.

[216] K. Shima, T. Kawamura, M. Yamaguchi, T. Yoshida, and T. Otsuka, “Implementation and Layout Optimization of Motion Sensor for Anomaly Detection in Daily Life of Elderly Persons,” in Proceedings of 2021 13th International Conference on Information and Communication Technology and System, ICTS 2021, 2021, pp. 140–145. doi: 10.1109/ICTS52701.2021.9608517.

[217] G.-S. Ham, M. Kang, S.-T. Joung, and S.-C. Joo, “Edge Computing-Based Medical Information Platform for Automatic Authentication Using Patient Situations,” KSII Transactions on Internet and Information Systems, vol. 17, no. 4, pp. 1049–1065, 2023, doi: 10.3837/tiis.2023.04.001.

[218] V.-T. Truong, A. Nayyar, and S. A. Lone, “System performance of wireless sensor network using LoRa-Zigbee hybrid communication,” Computers, Materials and Continua, vol. 68, no. 2, pp. 1615–1635, 2021, doi: 10.32604/cmc.2021.016922.