International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2024/32 Volume 11, Issue 4, July – August (2024)

RESEARCH ARTICLE

Enhancing Healthcare Monitoring with Efficient Computation Offloading in Fog Computing

Dinesh Anand
School of Computer Applications, Lovely Professional University, Phagwara, Punjab, India.
dinesh.researchca@gmail.com

Avinash Kaur
School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India.
✉ avinash.14557@lpu.co.in

Parminder Singh
School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India.
parminder.16479@lpu.co.in

Received: 22 May 2024 / Revised: 14 August 2024 / Accepted: 25 August 2024 / Published: 31 August 2024

Abstract – The exponential growth of produced data by healthcare monitoring devices poses a substantial challenge for conventional fog-based computing frameworks. Fog computing, a dispersed computing prototype that expands fog computing capabilities to the network's edge, emerges as a promising solution to address this challenge. This paper, proposes a technique for offloading computations for healthcare monitoring in fog computing, aiming to minimize task completion time, consumption of energy, execution time ratio and response time analysis. Enhancing Healthcare Monitoring with Optimal Computation Offloading in Fog Environment specifies that the research is focused on improving healthcare monitoring systems through the use of fog computing. In this approach, data processing is carried out closer to the source, such as medical devices or sensors, instead of depending only on centralized cloud servers. The "computation offloading" technique is moving computational workloads from less powerful devices to edge or fog nodes with more processing power. By using this method, the research seeks to improve real-time data processing, minimize latency, maximize resource use, and improve security in healthcare monitoring by retaining confidential data closer to its source. The goal of the study is to show how this strategy might result in healthcare monitoring systems that are more effective and efficient, especially when quick decisions and great data security are required. The proposed technique dynamically offloads computation tasks to fog nodes based on real-time network conditions, resource availability, and task characteristics. It emphasizes the achievement of superior performance metrics including the shortest job completion time, lowest energy consumption, and minimal cost compared to existing task offloading methods within healthcare contexts. The technique notably achieves a reduction of up to 31.1% in task completion time, 66.67% in energy consumption, and 20% in execution time ratio compared to existing task offloading methods in healthcare contexts. Additionally, it improves response time by 40%, demonstrating superior performance metrics. It conducts a thorough assessment of the proposed technique's effectiveness through key performance indicators such as Task Completion Time, Energy Consumption, Execution Time Ratio, and Response Time Analysis. Finally, a detailed comparative analysis against established techniques enriches the discussion, providing valuable insights into the superiority of the proposed technique.

Index Terms – Fog Computing, Healthcare Monitoring, Computation Offloading, Dynamic Task, Resource Optimization, Task Completion Time, Energy Consumption, Execution Time Ratio, Response Time Analysis.

1. INTRODUCTION

The swift procedure in healthcare monitoring technologies has resulted in an overwhelming growth in data produced by medical sensors, wearable devices, and patient-centric applications. This myriad of data presents a notable challenge for conventional cloud-based computing architectures, characterized by increased latency and constrained bandwidth. Fog computing, an emerging paradigm of computing distributed across multiple systems that expands cloud capabilities to the proximity of end-users, presents a captivating solution to tackle these issues [1]. By permitting data processing and analysis in real time at the edge of the network, fog computing defines the challenges posed by the data deluge in healthcare monitoring technologies. Currently fast pace of life and markedly improved living standards of people, has led to a growing recognition of the importance of addressing health concerns [2].

The healthcare sector has rapidly advanced with the emergence of the Internet of Things (IoT). The Internet of Medical Things (IoMT) seamlessly integrates human-centric

RESEARCH ARTICLE

data, contains medical history of the person, prescriptions, allergies, laboratory test outcomes, and personal matrices. This digital integration significantly grows the efficacy and quality of medical services. User data from different geographical areas can be centralized in a common data center, facilitating the sharing of medical information and transcending the temporal and spatial constraints of conventional healthcare models, all while safeguarding privacy. The IoMT promptly triggers alerts upon detecting anomalies in data. The IoMT enables the timely location, tracking, and monitoring of users [3]. Furthermore, the IoMT optimally delivers its economic benefits by minimizing intermediate steps and achieving optimal outcomes in the least possible frame of time, thereby providing patients with the most gratifying healthcare provisions. In recent times, the proliferation of IoT devices and widespread applications has experienced a substantial and consistent uptrend. Although, the swift expansion of communication technologies is keeping rapid pace to meet the growing demands [4].

Despite this, certain applications, particularly those with time-sensitive and crucial trust needs, necessitate specialized technologies. The concept of trust is critical in identification and pinpointing malicious entities masquerading as legitimate within the network. The most notable sectors for the expansion of IoT are medical services and healthcare facilities. The integration of IoT in healthcare mitigates costs and enhances the quality of user's lives by enabling the monitoring of everyday actions, incorporating sleep cycles, nutritional habits, and exercise regimes. This monitoring facilitates the generation of specific tips aimed at promoting a healthier lifestyle. Furthermore, the IoT application has proven benefits in various medical domains within the healthcare setting. This includes management of patient information, continuous real-time tracking, handling health emergencies, overseeing blood information, and overall health management [5]. The abundance of health information generated by IoT devices and sensors is collected, processed, and analyzed to provide valuable insights. Fog computing empowers healthcare systems to swiftly make intelligent decisions in emergency situations, particularly for time-sensitive healthcare problems like critical health condition due to COVID-19.

Additionally, it enhances data security by minimizing response times, surpassing other computing methods such as the cloud. The real-time processing capabilities, edge analytics, improved scalability, and enhanced security features make fog computing a compelling solution for addressing the evolving needs of modern healthcare systems. As the technology continues to mature and healthcare organizations embrace its potential, fog computing is ready to play an important role in creating the future of health-related delivery. Fog computing offers several advantages in relation to health-related issues is its caliber to provide real-time processing of data [6].

In emergency situations, such as a sudden surge in COVID-19 cases, quick and informed decision-making is paramount. Processing data at the edge is made easier by fog computing. ensuring timely analysis and actionable insights, which is crucial for effective healthcare interventions. Fog computing integrates edge analytics into healthcare systems, allowing for on-the-spot data analysis without the need to send large volumes of information to centralized cloud servers. The feature is particularly beneficial for resource-intensive applications like medical imaging and diagnostic tools, where reduced latency is essential for accurate and swift decision-making. Fog computing provides a scalable framework that is capable to cater the dynamic and evolving needs of healthcare systems. The volume of healthcare data regulated to grow at an increasing rate, fog computing allows for distributed computing resources that can scale up or down as per the requirement. This scalability feature helps the healthcare organizations can efficiently manage the tasks, optimizing both performance and resource utilization [7].

While fog computing offers substantial benefits in terms of real-time processing and scalability, addressing concerns regarding security and privacy is paramount in the healthcare sector. Fog computing mitigates some of these concerns by maintaining sensitive data near to the source, curbing the risk of unauthorized access during transmitting data to central cloud servers. Additionally, fog computing enabled localized safety precautions, for example employing encryption and controls over access, providing an extra layer of protection for healthcare data. Fog computing makes it easier to monitor patients in real time using sensors and wearable technology. At the network's edge, vital signs including blood pressure, glucose levels, and heat rate can be continuously monitored. This enables healthcare providers to receive immediate alerts and make timely interventions in cases of emergencies, ensuring patient safety. Fog computing enhances the processing of medical imaging data like CT scans, MRIs, and X-rays at the edge of the network. This reduces the time required for image analysis and interpretation, leading to quicker diagnosis and treatment planning. It also alleviates the burden on centralized servers, optimizing overall system performance [8]. Smart drug delivery systems integrated with fog computing can monitor patient adherence to medication regimens. Sensors on medication packaging can send data to the edge, ensuring that patients take medications as prescribed.

In case of non-compliance, healthcare providers can be alerted in real-time, allowing for timely intervention. Fog computing tackles security and privacy issues in healthcare by maintaining sensitive patient data closer to its origin. Localized processing ensures that critical information stays

RESEARCH ARTICLE

within the healthcare facility, reducing the risk of unauthorized access during data transmission to centralized cloud servers. This decentralized approach enhances data security and privacy. Fog computing aids in the efficient management of healthcare facilities by integrating data from various sources, such as patient records, equipment sensors, and environment monitoring devices. This enables real-time analysis of facility operations, optimizing resource utilization, and ensuring a safe and comfortable environment for patients and staff. Fog computing supports the development of personalized healthcare applications by processing data from wearable devices and patient history at the edge. This enables the delivery of tailored health insights and recommendations, fostering patient engagement and empowerment in managing their health proactively [9].

Fog computing in the healthcare sector enhances the overall performance of healthcare delivery by providing real-time processing, improving data security, and assisting comprehensive applications that contribute to better patient results and healthcare management. As technology continues to progress, the incorporation of fog computing is anticipated to have a growing significance in influencing the future of healthcare. In the realm of fog computing, diverse vendors are actively participating in providing services utilizing fog computing for a multitude of concerns. Major providers of cloud service are extending their offerings to the periphery of user locations to enhance effectiveness. Private cloud owners are offering their underutilized resources for lease to local businesses. Environment utilizing fog computing is impacted by different internet service providers and mobile network operators responsible for information exchange. This dynamic involvement of multiple entities introduces a new challenge known as the Necessity for Trust among fog nodes. Fog nodes in healthcare contribute significantly to improving operational efficiency, adding innovation and patient care in healthcare solutions, all while addressing the distinctive challenges faced by the industry [10]. Given the sensitivity of healthcare data and the rigid laws such as the Health Insurance Portability and Accountability Act (HIPAA), fog computing enabled local processing of sensitive data at the edge. This boosts safety and confidentiality by mitigating the necessity of transmitting data over external networks.

In recent times, there has been a notable surge in the use of smart devices. The complexity and demand of applications capable of performing intricate tasks have also risen sharply. Examples of such applications include video streaming, live gaming, and face recognition, all of which heavily utilize various mobile resources such as CPU, memory and battery. Despite the portability, accessibility and affordability of mobile devices, many struggle to last a full day due to the intensive use of resources, particularly the CPU. Mobile Cloud Computing has surfaced as a resolution to improve the execution and battery life of mobile devices by integrating network, mobile and cloud computing technologies. This involves leveraging the abundant resources of cloud service providers. Fog computing supports the deployment of clinical decision support systems within hospitals. These systems leverage patient data, medical knowledge and algorithms to help medical professionals in building informed decisions about care of patient. Fog nodes enable the fast execution of CDSS (Clinical Decision Support System) algorithms, providing timely recommendations at the point of care. Fog nodes enable hospitals to offer telemedicine services and remote consultations to patients. By deploying fog nodes within hospital networks, healthcare providers can securely transmit patient data, conduct virtual appointments, and collaborate with specialists in real-time, regardless of geographical distance [11]. Fog computing assists hospitals optimize resource utilization and reduce operational costs. By distributing computational tasks between fog nodes and centralized servers, hospitals can minimize network latency, conserve bandwidth, and ensure continuous access to critical healthcare services [12].

1.1. Problem Statement

This approach seeks to provide a comprehensive understanding of the performance and advantages offered by our proposed technique in the context of healthcare monitoring within cloud computing environments. Our proposed technique tackles the challenge of resource constraints encountered in fog devices. Given the limited resource capacity of these devices, it is crucial to allocate cloud resources efficiently to ensure optimal application throughout. To address this, a proposed technique for resource scheduling is introduced. Initially, a regression algorithm is employed to reformulated user requests, reducing the occurrence of repeated requests. The paper presents a method for transferring healthcare monitoring computations to fog computing, with the goal of reducing task completion, energy consumption, execution time ratio, and analyzing response times. It also compares results of our proposed technique with existing techniques.

1.2. Contributions

The contributions of this paper are:

1. Real-Time Data Operation: The capability of fog computing to handle and analyze healthcare data instantaneously, decreasing latency and enabling rapid decision-making is showcased.

2. Enhanced Security and Privacy: The sensitive data put closer to the source results that fog computing improves data security and privacy. It also lowers risks associated with data transmission.

3. Scalability and Efficiency: The scalable design of fog computing, which can be adjusted to meet the changing

RESEARCH ARTICLE

demands of healthcare systems, optimizing performance and resource use is highlighted.

4. Emergency Response: The role of fog computing in managing healthcare emergencies, such as COVID-19 outbreaks, by providing prompt and actionable insight is investigated.

5. Innovative Healthcare Applications: The aggregation of fog computing into various healthcare applications, including telemedicine, patient monitoring, medical imaging, and smart drug delivery systems is explored.

1.3. Objective

To introduce a heuristic algorithm aimed at smartphones to determine whether to accept computation tasks from wearable devices and subsequently offload them to cloud servers. The ultimate goal is to devise a calculative offloading method that enhances the power efficiency of health-related devices.

1.4. Structure of the Research Paper

The remaining portions of the paper are arranged as follows: Section 2 pertains to explanation of the related work with respect to healthcare monitoring and for optimal service offloading within the context of fog computing. Section 3 proposes a heuristic algorithm aimed at smartphones. Section 4 discusses the results obtained after conducting the carefully designed experiments. By the end, the conclusion of the paper is presented in Section 5.

2. RELATED WORK

Currently, the demand for healthcare assistance is growing, leading to a strain on resources with fewer healthcare professionals available to handle the increasing number of patients. Management of patient data effectively is crucial, especially when it comes to monitoring standard test outcomes, which can become tedious for doctors, even if the outcomes are normal. For this challenge, IoT technology suggests a promising outcome by enabling real-time remote monitoring of patients through sensor-equipped devices. Nonetheless, the immense amount of data produced by IoT devices can inundate conventional systems, leading to delays when transferring data from the cloud to the application. To curb this concern, the author has proposed leveraging fog computing as an intermediary solution. By utilizing fog computing, the author has processed data closer to its source, mitigating latency and enabling instantaneous monitoring. Moreover, the author has incorporated notification systems and machine learning algorithms to enhance the prediction procedure, leveraging various inputs to provide insightful analysis of patient test results in a simpler and cost-effective manner.

Menna et al. [1] discussed that the IoT has revolutionized the method that interacts with technology by connecting various devices to the internet, which leads to the generating vast amount of data on a regular basis. This data explosion is fuelled by the increasing adoption of IoT devices such as smart wearable, smartphones, and the development of smart cities. Among the various applications of IoT, healthcare stands out prominently, where IoT devices play a vital role in remote patient health monitoring and management. Fog computing has become as a promising alternative architecture to tackle these demands. By bringing computational resources close to the network's edge, fog computing broadens the concept of the cloud and enables data processing and examination to occur closer to where the data is produced. The author has proposed a novel approach called Trust Enforced Computation offloading technique (TEFLON) specially designed for securing and trustworthy applications using fog computing. TEFLON constitutes of two key algorithms: maximum service offloader and evaluation of trust. These algorithms are aimed at dealing with trust and security concerns while mitigating time of response. Through extensive studies for simulation, outcomes demonstrate that the network of TEFLON significantly shows an upward trend in the rate of success for fog collaboration reduces average time for applications that are susceptible to delays and enhances the overall trustworthiness of fog-based healthcare applications. By addressing the confidence and security challenges inherent in fog based computing environments, TEFLON paves the way for the widespread adoption of fog-based computing in sensitive areas such as medical locations.

The proposed TEFLON framework aims to enhance trustworthiness in fog computing environments, particularly for sensitive applications. This framework introduces novel algorithms for optimal service offloading and trust assessment, reducing response times and improving success rates for fog collaborations. The subsequent discussion delves into related works and research efforts aimed at addressing trust and computational offloading challenges in fog computing. Various approaches such as commitment-based trust assessment, hierarchical trust systems, and load-balancing techniques are explored. Additionally, trust management protocols, reputation-based models, and workload allocation strategies are discussed to enhance the efficiency, security, and reliability of fog based computing systems. These efforts underscore the importance of trust in mitigating security risks and optimizing resource utilization in fog computing environments, particularly for delay-sensitive and trustworthy applications.

Aazam et al. [2] tells the burgeoning importance of IoT devices and their associated applications, highlighting the exponential growth in data generation. It emphasizes increasing utilization of IoT devices such as wearable, smartphones, and systems for smart cities, with healthcare emerging as a primary domain for IoT applications. Given the sensitive nature of healthcare data and the need for

RESEARCH ARTICLE

computational operations like storage and analytics, there is a demand for robust environments to support remote patient monitoring. Traditional cloud environments face challenges in meeting privacy and accessibility requirements for such data. In response to these challenges, fog computing has developed as a promising architecture. Fog computing facilitates computation, storage, control, and network services near users, addressing issues related to data privacy and accessibility.

Qiu et al. [3] discussed that IoMT has played a vital role in managing the spread of the virus during the COVID-19 pandemic. It facilitates various tasks such as diagnosing and treating patients across different levels of healthcare facilities, promptly separating and caring for individuals who have been confirmed or are suspected cases, and preventing the transmission of the novel coronavirus. This study focuses on optimizing healthcare monitoring within IoMT deployed on fog computing (FOGC-IoMT). The optimization issue is complex, non-convex, and nonlinear, involving Quality of Service (QoS) requirements, power limitations, and mobile front haul constraints. To address this, the author proposed a scheme that decouples the problem into three independent subproblems, enabling efficient offloading of computation and management of resources in FogC-IoMT. The simulation results demonstrate that the proposed optimization network is effective in cost minimization. The investigation of the author centers on mitigating the cost usefulness for medical users in fog computing-based IoMT healthcare monitoring. The author considers various factors such as energy consumption, transmission delay, QoS requirements, wireless front haul constraints, and power limits.

To tackle the nonlinearity and non-convexity of the initial problem, the author decomposed it into three different sub-problems: sub-channel power allotment, assessment, and offloading for medical computation. A low-complexity algorithm to address each sub-problem individually, enhancing solution efficiency was introduced. Matching theory to allocate sub-channels between Fog Accesses Points (F-APs) and medical users (F-MUs) using Non-Orthogonal Multiple Access (NOMA) employed. Moreover, the transformation the non-convex fractional power assignment problem into a sub-tractive form has done to facilitate its resolution. The outcomes of simulation validate that the effectiveness of the proposed is less than the ideal, simple schemes for offloading for health-related computation and allocation of resources in FogC-IoMT. Furthermore, the author has extended the investigation to cut back cost optimization for all F-MUs in FogC-IoMT while considering multiple constraints. By decoupling the problem into medical task offloading, wireless front haul bandwidth allocation, sub-channel assignment, and power allocation sub-problems, the author proposed a suboptimal, low-complexity scheme to mitigate consumption of energy in health-related monitoring within FogC-IoMT. The author has suggested that future research should delve into enhancing emotion-aware capabilities in FogC-IoMT to provide more comprehensive patient-centered medical services that offer several advantages over traditional cloud-based approaches [13, 14].

Computational offloading has garnered significant attention among researchers. Smart wireless healthcare devices introduce crucial challenges such as battery longevity, processing speed, and offloading costs, which hinder the widespread adoption of wearable technology [15]. Over the past few years, a wealth of research has focused on offloading computational tasks from healthcare devices. In the realm of computing for the cloud, Deep Learning (DL) techniques have been extensively utilized. Simultaneously, considerable attempts have been directed towards mitigating the burden of DL works to render them reliable on healthcare devices. However, wearable devices, with their limited computational resources compared to smart healthcare devices, face significant hurdles in executing DL tasks. Integrating deep learning techniques into fog nodes enables distributed, edge-based intelligence, empowering healthcare devices to make context-aware decisions in real time while leveraging the scalability and processing ability of cloud resources when needed [16]. DL inference tasks closer to the data source, latency is mitigated, making decisions in real-time feasible for time-critical applications like healthcare monitoring and high workload automation. Additionally, with the help of deep learning, transmitting raw data to a server of remote cloud for processing can be bandwidth-intensive, especially for applications generating large volumes of healthcare data. By implementing data locally on nodes of fog, only relevant insights or aggregated results need to be sent to the cloud, conserving bandwidth. DL techniques in fog nodes utilize deep learning algorithms and models within the fog computing paradigm. Computing related to fog expands computing related to the cloud to the network's periphery, enabling storage and computation, and networking resources close to where data is generated and consumed, thereby reducing latency and bandwidth usage. Integrating deep learning into fog nodes enables the processing and analysis of data locally. In scenarios where intermittent connectivity or network disruptions occur, fog nodes equipped with deep learning capabilities can continue to perform inference tasks, ensuring uninterrupted operation [17].

Daraghami et al. [18] created an architecture for remote health vital sign monitoring that consists of three layers: Edge, Fog, and Cloud. To increase security while preserving quick calculation and transmission times, the study also investigated several authentication techniques. The suggested design decreased the execution time by an average of 38.5%, the authentication time by 35.1%, and the NB-IoT latency by 59.9% for many devices.

RESEARCH ARTICLE

Premlatha et. al. [19] suggested that for efficient IoT-Fog computer network job offloading, the "Optimal Energy-efficient Resource Allocation (OEeRA)" approach, is based on the "Minimal Cost Resource Allocation (MCRA)" and "Fault Identification and Rectification (FIR)" algorithms. Each device is to have minimum one Fog Node and Resource Block assigned to it by the MCRA algorithm, which also makes sure that each Fog Node is connected to required number of Resource Blocks and devices. According to the FIR algorithm's proposal, the remaining Resource Blocks are gathered and kept as reserve to replace the defective Reserve Blocks, resulting in improved processing and response times and increased fault detection accuracy. By adjusting FN, RB, and IoT devices, the energy efficiency of the suggested "OEeRA" method is calculated using the "MCRA and FIR" algorithms. As per the performance appraisal, the suggested algorithm could attain the highest energy efficiency of 6.12 × 109 bit/J, 5.69 × 1010 bit/J, and 3.019 × 1010 (bit/J) for different Fog Nodes, IoTs, and Resource Blocks.

The authors have proposed an offloading scheme comprising assessment, selection, scanning, and offloading phases, healthcare devices via Bluetooth for energy-intensive tasks, and providing a communication and computation healthcare model for decision-making [20,21,22]. Another technique divides tough tasks with highest power utilization into process for offloading via Bluetooth and communication for Wi-Fi, comparing utilization of energy using Million Floating Point Operations (MFLOP). In performance analysis, higher MFLOP values indicate greater computational throughput, implying that a healthcare system can process more data or perform more complex computations within a given time period. As such, MFLOP is an essential metric for assessing the computational efficiency and scalability of computing systems, particularly in field of healthcare systems. MFLOP provides a standardized measure to compare the computational capabilities of different hardware platforms or to evaluate the efficiency of algorithms in terms of computational complexity. Floating-point operations involve mathematical computations using numbers represented in floating- point format, which allows for a wide range of values and precision. Healthcare analytics analyzes large-scale healthcare datasets to derive insights into patient outcomes, population health trends, and healthcare resource utilization. MFLOP measures the computational complexity of statistical analysis, data mining methods, algorithms used in machine learning applied to healthcare data, facilitating efficient data processing and knowledge discovery. Computational models and simulations play vital role in drug discovery and development processes, including virtual screening, molecular docking, and pharmacokinetic modeling. MFLOP can quantify the computational workload of these simulations, enabling researchers to optimize algorithms, parallelize computations, and accelerate the drug discovery pipeline. MFLOP is used to evaluate the computational efficiency of these algorithms and optimize their performance for real-time clinical decision-making. It quantifies computational complexity of these bioinformatics tasks, aiding in the design of efficient algorithms and high-performance computing infrastructures for genomics research and personalized medicine applications.

Additionally, works explore energy minimization in local computation and offloading, latency performance in large-scale networks, and workload distribution in hierarchical cloudlet architectures. Despite advancements, attention to legitimate communication module selection remains limited. For comprehensive insights into cloud computing, studies delve into healthcare devices' efficiency, categorization, and the fundamental shape of offloading techniques. Table 1 given below presents the summary of related work.

Table 1 Summary of Related Works

Reference	Contribution	Remarks
[1]	The author has proposed a novel approach called Trust Enforced Computation offloading technique for trust worthy applications using fog computing	Outcomes demonstrate that the network of TEFLON significantly shows upward trend the rate of success for fog collaboration, reduces average time for delay-sensitive applications and enhances the overall trustworthiness of fog-based healthcare applications.
[2]	Highlighting the exponential growth in data generation.	Fog computing facilitates computation, storage, control, and network services closer to users, addressing issues related to data privacy and accessibility.
[3]	This study focuses on	Outcomes of simulation demonstrate the

RESEARCH ARTICLE

	optimizing healthcare monitoring within IoMT deployed on fog computing (FOGC-IoMT).	efficacy of the proposed optimization network in terms of cost minimization
[18]	A remote health vital sign monitoring system with three layers of computing infrastructure: Edge, Fog, and Cloud.	The recommended architecture reduced the NB-IoT latency by 59.9%, the execution time by 38.5% on average, and the authentication time by 35.1%.
[19]	The OEeRA approach is based on the "Minimal Cost Resource Allocation" (MCRA) and "Fault Identification and Rectification" (FIR) algorithms	The proposed algorithm was able to attain the highest EE of 6.12 × 109 bit/J, 5.69 × 1010 bit/J, and 3.019 × 1010 (bit/J) for different FNs, IoTs, and RBs

3. PROPOSED MODEL

In contrast to servers for cloud with virtually unlimited computing resources, smartphones are constrained by factors such as cell efficiency. As a result, continually handling offloading requests from wearable devices solely on smartphones poses usability challenges. This paper introduces a heuristic algorithm aimed at smartphones to determine whether to accept computation tasks from wearable devices and subsequently offload them to cloud servers [23,24]. The ultimate goal is to devise a calculative offloading method that enhances the power efficiency of health-related devices.

Current offloading methods generally follow two workflows. Firstly, complex computations with high energy consumption are offloaded to cloud servers, which then return the results. Secondly, smartphones act as intermediary servers, facilitating the transmission of operations from wearable devices to cloud servers. However, the suggested technique incorporates a process of making decision for smartphones to evaluate whether to accept wearable device operations, considering smartphone energy efficiency, and whether to subsequently offload tasks to cloud servers [25].

Through the Task Analyzer, tasks are evaluated to determine whether they should be offloaded to smartphones based on energy consumptions considerations. Subsequently, smartphones make informed decisions about task acceptance and potential offloading to cloud servers, taking into account factors like remaining battery energy and communication efficiency. Following this method, the wearable gadget offloads the task of computation to the smart device and awaits the outcome.

This approach enables smartphones to dynamically manage computation tasks from wearable devices, optimizing energy usage and improving overall system performance. By leveraging cloud resources when appropriate, the aim is to overcome the limitations posed by smartphone resources and enhance the usability and energy efficiency of smart devices in offloading scenarios. Figure 1 shows the workflow diagram of the proposed technique.

3.1. Offloading Decision Model

There are several factors involved in determining whether healthcare devices should offload their computations, including the local consumption of energy needed for processing and the Wi-Fi power costs incurred when offloading tasks to a cloud server. Firstly, the paper outlines the Millions of Dhrystone Instructions Executed per Second (MDIES), as detailed in prior research. Secondly, it presents healthcare decision models for computation offloading that take into account the communication energy costs associated with offloading from each smart device's perspective.

3.2. MDIES-Based Workload Model

MDIES, which stands for Millions of Dhrystone Instructions Executed per Second, quantifies the CPU's performance by measuring the number of instructions it can process per second.

$$W = \frac{MDIES_m}{U_{mdies}} \times U_{task} \times t_{task} \quad (1)$$

In equation (1), The workload W_L is obtained by the product of $MDIES_m$ per CPU utilization and task execution U_{task} and multiplied by time taken to execute the task t_{task} when the Dhrystone Instructions are applied, and $MDIES_m$ is the median.

3.3. Healthcare Device's Decision Model

The decision to offload tasks from a healthcare device hinges on a comparison of the power usage of nearby computing on the healthcare device with that of offloading the task to a healthcare device via wireless device. Equation (2) is utilized to quantify the energy expended when the computation task is processed locally on the healthcare device.

$$E_{Hd} = E_{WL\_healthcare} \times W_L \quad (2)$$

In equation (2), the energy consumption of the healthcare device is computed by multiplying the energy per unit workload by work load.

$E_{WL\_healthcare}$ represents the energy expended per healthcare instance of the healthcare device, while W_L signifies the workload achieved based on the workload framework. When the healthcare gadget delegates the computation task to the wireless device, the energy consumption of the healthcare device ($E_{offloading\_healthcare}$) comprises the energy expended for wireless communication ($E_{wireless}$) and the waiting duration until the outcome is collected from the healthcare device ($P_{wait\_healthcare} \times t_{wireless}$).. $E_{offloading\_healthcare} = P_{wait\_healthcare} \times t_{fog\ device} + E_{wireless\ device}$ (3)

In equation (3), the energy consumption of the healthcare task $E_{offloading\_healthcare}$ is obtained by summing the product of power consumption during waiting time and the time spent on fog device with energy consumed by wireless device.

Equation (3) represents the energy price framework of the healthcare device when computation is offloading to the fog device.

$$E_{wireless\ device} = \frac{Pwirelessdevice(power)}{Twirelessdevice(throughput)} \times N\ (\text{data size}) \quad (4)$$

In Equation (4), the energy consumption of wireless device is calculated by dividing the power consumption of the wireless device by its throughput and multiplying the result by data size N.

$E_{wireless\ device}$ represents the energy expenditure attributed to wireless communication for the healthcare device. Within this study, a power cost framework specifically tailored to account for wireless communication energy costs was formulated.

$$E_{wifi} = \frac{Pwifi}{Twifi} \times N \quad (5)$$

In equation (5), the energy consumption of the Wi- Fi connection is determined by dividing the power consumption of P_{wifi} by its T_{wifi} and then multiplying this result by data size N.

The decision to offload computation tasks can be made by comparing the energy consumption ($E_{healthcare\ device}$) when the healthcare device performs computations locally versus the energy consumption ($E_{offloading\_healthcare}$) when offloading tasks to the fog device. This comparison can be expressed as done in equation (6).

Offloading if $E_{healthcare\ device} > E_{offloading\_healthcare}$ (6)

Non-offloading if $E_{healthcare\ device} \leq E_{offloading\_healthcare}$

3.4. Fog Devices Decision Model

Upon receiving an inquiry for offloading from a healthcare device, the fog device initially evaluates whether to accept the request. This evaluation involves computing the energy consumption (E_{device}) associated with locally processing the requested operation and then comparing it against the outstanding energy stored in the battery ($E_{battery}$). Equation (7) encompasses the energy expenditure for wireless interaction required to transmit the outcome of the computed activity back, which represents the energy consumed when the fog device manages the computation assignment nearby as requested by the healthcare device.

$$E_{device} = W_{L\_device} \times W_L + E_{wireless\ device} \quad (7)$$

In equation (7), energy consumption of a device can be obtained when the energy consumed by device due to its work load ($W_{L\_device} \times W_L$) is added to the energy consumed by its wireless operations.

$$E_{battery} = B_{current} \times V \times 3600s \quad (8)$$

Equation (8) is employed to assess if the fog device has the capability to offload, achieved by transforming the remaining battery capacity of the fog device into an energy unit. Furthermore, the fog device evaluates whether to acknowledge the request from the healthcare device. It quantifies and contrasts the energy utilization associated with offloading to the fog server. Equation (9) computes the energy usage ($E_{offloading\_device}$) when the fog device delegates the request from the healthcare device to the fog server and awaits the reception of results ($P_{wait\_device} \times t_{fog}$). Wireless device energy ($E_{wireless\ device}$) and wireless energy ($E_{wireless}$) are expanded for transmitting back to the healthcare device and fog devices, respectively.

$$E_{offloading\_device} = P_{wait\_device} \times t_{fog} + E_{wireless\ device} \quad (9)$$

To enhance the energy efficiency of fog devices, the decision to offload is determined by two comparisons as given in equation (10) and (11).

Offloading if $E_{healthcare\ device} < E_{device} \leq E_{battery}$

Non-offloading if $E_{device} \leq E_{battery} \leq E_{offloading\_healthcare}$ (10)

Offloading if $E_{offloading\_healthcare} < E_{battery} < E_{device}$

Refusal if $E_{battery} < E_{device} \leq E_{offloading\_healthcare}$ (11)

Furthermore, if local processing is not feasible but the energy cost ($E_{offloading\_device}$) of offloading to the fog server is lower than the remaining battery energy; the fog device offloads the task to the fog server once. However, if the energy consumption during offloading exceeds the remaining battery energy, the fog device declines the request from the healthcare device.

Algorithm 1 illustrates a health-related system integrating cloud computing, fog computing, and IoT devices. It delineates the distribution of tasks among these components. IoT health-related devices collect data from patient, which may include information from health sensors and wearable gadgets as per Figure 1. Then, Fog devices receive the acquired data via transmission. Data from IoT devices is analysed by fog devices. They process the data locally or forward it to the cloud for more intricate computations.

IoT_Healthcare_Device:

Step 1: Task Reception at IoT Healthcare Device

RECEIVE task at IoT_Healthcare_Device

Step 2: Decision at IoT Healthcare Device

task_analysis = Task_Analyzer(task)

IF Offloading_Manager_Decision(task_analysis)== "Offload" THEN

// Task is offloaded to Fog Devices

GOTO Fog_Device_Processing

ELSE

// Task is processed locally at IoT Device

result = Computation_Part(task)

Output_Manager(result)

END

END IF

Fog_Device_Processing:

Step 3: Task Offloading to Fog Devices

IF Fog_Device_Accepts(task) THEN

IF Fog_Device_Offloading_Decision(task) == "Offload to Cloud" THEN

// Task is offloaded to Cloud Servers

GOTO Cloud_Server_Processing

ELSE

// Task is processed locally at Fog Device

result = Computation_Part(task)

Output_Manager(result)

END

END IF

ELSE

// Task is refused or returned to IoT Device

REFUSE or RETURN task to IoT_Healthcare_Device

END

END IF

Cloud_Server_Processing:

Step 4: Task Offloading to Cloud Servers

result = Computation_Part(task)

Output_Manager(result)

END

Algorithm 1 The Proposed Computational Offloading Model

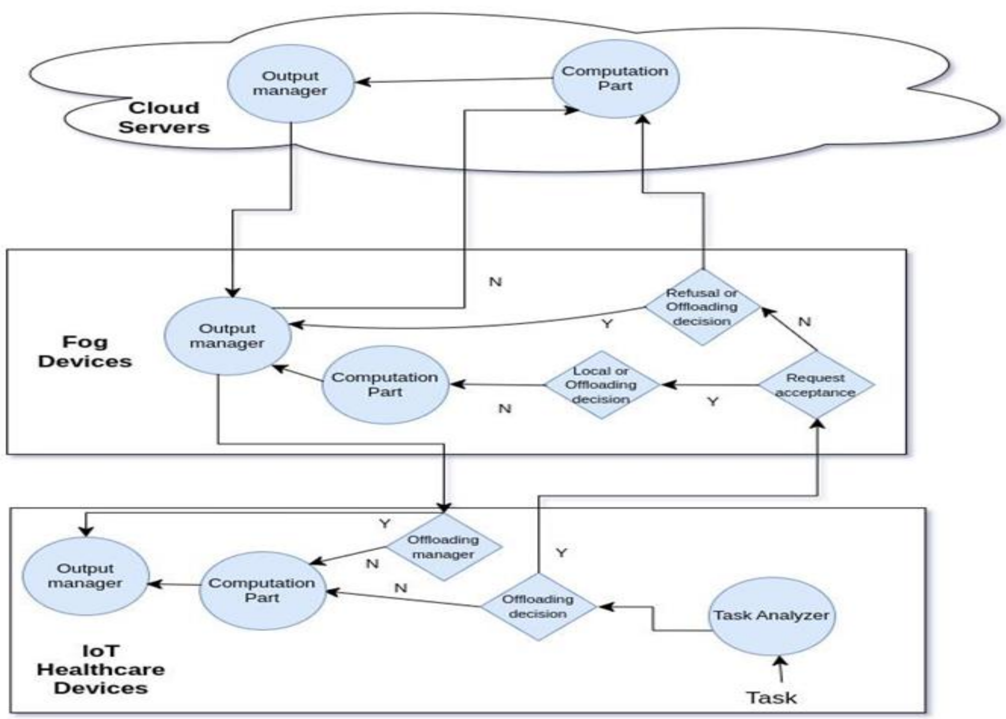

Figure 1 Workflow of the Proposed Offloading Model

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2024/32 Volume 11, Issue 4, July – August (2024)

RESEARCH ARTICLE

Determining factors include the complexity of the task and the volume of data. The task analyzer evaluates the data from fog devices and makes decisions regarding its processing. It assesses whether the task can be managed by a fog device or needs to be offloaded to the cloud. Cloud servers possess higher processing capabilities to manage intricate tasks. Final results are delivered to the appropriate destination depend on the intended use.

4. RESULTS AND DISCUSSIONS

In this part, carefully designed experiments were conducted to collect data, manipulated, and scrutinized to uncover insights and draw conclusions. Various methodologies, including controlled trials and statistical analysis, has been used to guarantee the authenticity and dependability of the results. The proposed method's performance was assessed by measuring and contrasting it with established collaboration of fog's benchmarking algorithms using health-related applications dataset.

4.1. Simulation Parameters

The suggested MDIES framework has been implemented through simulation using MATLAB on a system that has 16 GB RAM and an Intel Core i7 processor. Table 2 contains information on the simulation parameters.

Table 2 Simulation Variables

S.No	Simulation Parameter	Value
1	Simulation Environment	Matlab
2	Quantity of Fog nodes	50
3	Topology of the Network	Mesh Network Structure
4	Data Set	Heart Disease
5	Instances Tested	1988
6	Quantity of Attributes	13
7	Type of Data	Multidimensional
8	Data Transfer Rate	64 Mbps
9	OS	Windows 11
10	Fog Central GrowthCapability	2.4 GHz
11	RAM	16 GB

4.2. Dataset

This set of data originates from 1988 and comprises four databases: Long Beach V, Hungary, Switzerland, and Cleveland. It encompasses 76 attributes, with one being the predicted attribute, although published experiments typically utilize a subset of 13 attributes [25]. The patient's heart problem is indicated in the "target" field with values of 0 indicating no disease and 1 indicating disease.

4.3. Result Analysis

This section includes the tests that were carried out and concludes with a comparison of the suggested approach. To demonstrate the effectiveness of the algorithm, its performance is juxtaposed with other offloading algorithms like GA-ACO, CMS-ACO and FOTO.

The time for completion of task is influenced by the complexity of task scheduling and resource allocation. When contrasted with existing algorithms, the proposed algorithm demands less time for computation, as illustrated in Figure 2. Time for completion varies depending upon the nature of the job. The proposed algorithm accomplishes each task formulation and decision-making, thereby utilizing a shorter timeframe. The chart suggests that GA-ACO receives the highest number of requests, trailed by CMS-ACO, FOTO and proposed technique. As the number of requests show upward trend, there is a corresponding increase in total competition time for all four techniques. It seems that proposed technique has the shortest total completion time when compared to the other three techniques.

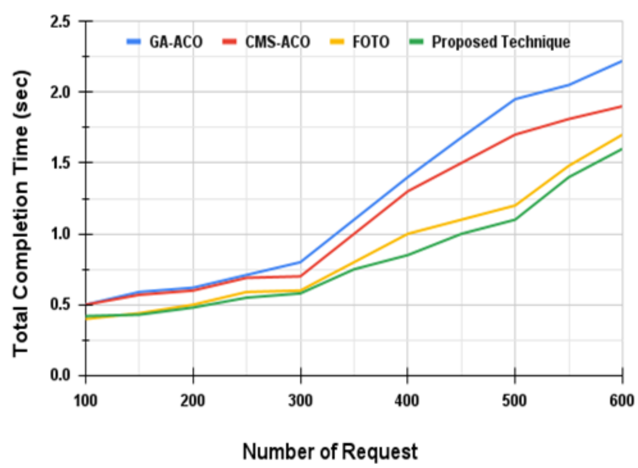

Figure 2 Task Completion Time

Figure 3 illustrates the average energy consumption for every type of task across four different techniques. Our proposed technique demonstrates the lowest energy consumption, averaging about 1Kwh per task. FOTO shows an average consumption of around 3 kWh per task. CMS-ACO consumes approximately 5 kWh per task. GA-ACO records the highest energy consumption, averaging around 7 Kwh per task. The proposed technique shows a reduction in energy consumption of 66.67% compared to FOTO, 80% compared to CMS-ACO, and 85.71% compared to GA-ACO.

RESEARCH ARTICLE

In Figure 3, the performance metrics of the proposed technique are showcased in comparison to existing algorithms. Proposed technique employs a multi-objective approach to optimize resource allocation within cloud data centers, particularly in identifying the optimal host for task offloading.

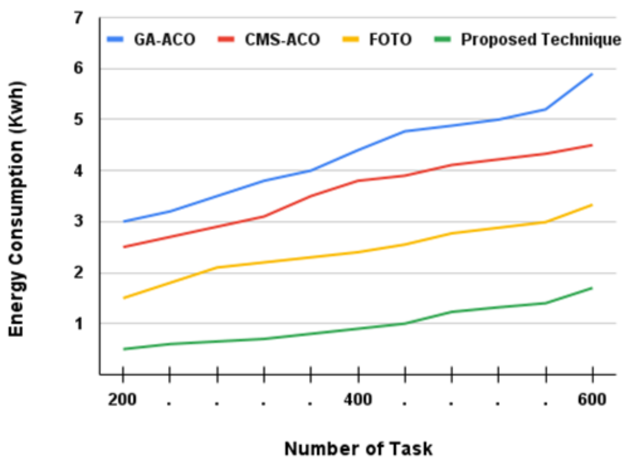

Figure 3 Energy Consumption

Notably, proposed technique demonstrates shorter response times and lower energy consumption compared to existing algorithms. This is achieved by allotment of virtual machines to hosts only when the required capacity matches the available resources. These findings show that the algorithm works better than the current ones in terms of resource usage, task completion time and efficiency of energy. These findings highlight the practical advantages of adopting proposed technique in cloud computing environments, where optimization of resource allotment and mitigating consumption of energy are crucial factors for enhancing overall system performance and sustainability.

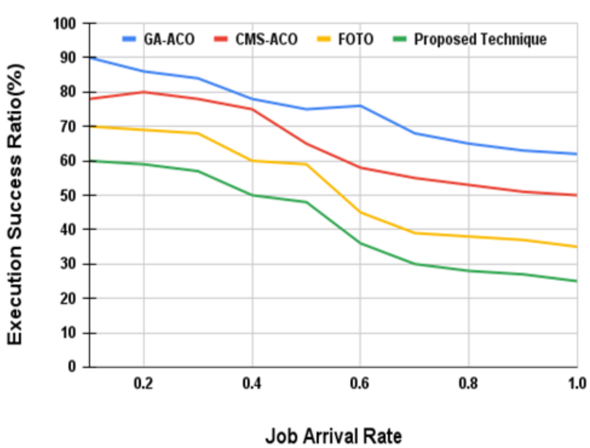

Figure 4 Execution Time Ratio

Figure 4 depicts the relationship between the Execution Success Ratio and average job arrival rate for four scheduling algorithms: GA-ACO, FOTO, CMS-ACO, and the blue, yellow, red, and green lines indicate the proposed technique. The Execution Success Ratio indicates the proportion of successfully completed jobs by each algorithm. Notably, the Proposed Technique demonstrates the highest average job arrival rate across all levels of Execution Success Ratios, suggesting its superior performance in job scheduling compared to the other algorithms. As the Execution Success Ratio increases, the average job arrival rate also tends to rise for all algorithms. This increase in the arrival rate appears to follow a consistent pattern across all algorithms. The Proposed Technique consistently exhibits the lowest average job arrival rate among all the algorithms, regardless of the Execution Success Ratio. At an Execution Success ratio of 0.6, the Proposed Technique job arrival rate ranges from low of 0.1 to a high of 1.0. This pattern is consistently observed across Execution Success ratio values of 0.7, 0.79 and 0.9 when compared to FOTO, CMS-ACO, GA-ACO, the proposed Technique superior performance reinforces its relative efficiency.

The figure 4 provides insights into the analysis of execution time. By changing the quantity of cloudlets, the execution time is assessed. The analysis reveals that the execution time decreases even when a greater number of cloudlets are utilized. Conversely, a smaller number of cloudlets necessitate longer execution times. Notably, the proposed technique demonstrates significantly improved execution time compared to existing approaches, indicating its efficiency in managing computational tasks within the cloud environment.

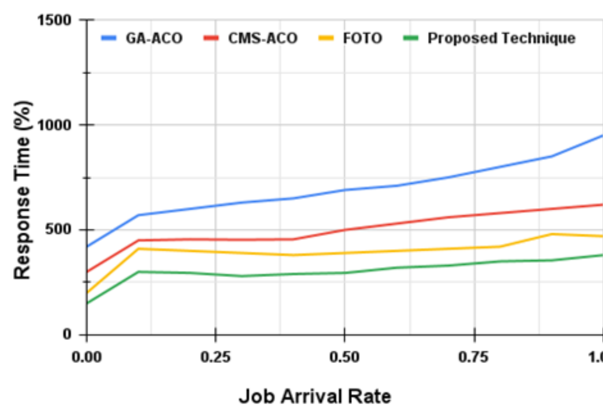

Figure 5 Response Time Analysis

Figure 5 illustrates the job arrival rate concerning three distinct scheduling algorithms, GA-ACO, CMC-ACO, FOTO, with proposed technique. The y-axis denotes the job arrival rate, whereas the x-axis represents the response time. At GA-ACO, the response time is observed to increase significantly

RESEARCH ARTICLE

with the job arrival rate. It starts at around 450% and rises sharply, reaching around 950% at the highest job arrival rate. At CMS-ACO, the response time is seen to increase with the job arrival rate but at a slower pace compared to GA-ACO. It starts slightly below 350% and increases steadily reaching about 625%. At FOTO, A relatively moderate increase in response time is shown by this technique. It starts at around 200%, increasing gradually and reaching just above 450%. The lowest response time across all job arrival rates is exhibited by the proposed technique. It starts at about 160% and rises slightly, maintaining a response time just below 370%. The proposed technique is consistently outperformed by the other methods in terms of response time, indicating its efficiency in handling job arrival rates.

The job arrival rate signifies the average influx of jobs into the system per unit of time. From the figure, it is evident that the GA-ACO algorithm generally exhibits a higher job arrival rate compared to the CMS-ACO algorithm across various response times. Response time refers to the duration taken to complete a job from its arrival at the system.

5. CONCLUSION

The work presented here aimed at reducing energy consumption, minimizing task completion time, and lowering data center costs by offloading tasks from healthcare devices to cloudlets. The proposed technique restructures user queries and identifies accepted requests using a linear regression framework. In a cloud data center, it uses multi-objective functions to choose the best host. The method's effectiveness is evaluated through the simulator and contrasted with task offloading algorithms such as CMS-ACO, FOTO, and GA-ACO. The simulation results demonstrate that the suggested technique remarkably reduces energy consumption, decreases task completion time, and cuts data center costs. It is particularly applied in healthcare devices using Fog computing. The execution time was determined by adjusting the quantity of cloudlets. and the technique was found to outperform existing ones. Looking ahead, plan was made to develop new fault tolerance mechanisms to manage virtual machine failures and explore more intricate task offloading relationships to further improve energy savings for healthcare devices. Additionally, security measures will be implemented to ensure the protection of offloading tasks.

REFERENCES

[1] Meena, V., Gorripatti, M., & Suriya Praba, T," Trust enforced Computational offloading for health care applications in fog computing", Wireless Personal Communications, Vol.119, no. 2, pp. 1369-1386, 2021.

[2] Aazam, M., Zeadally, S., & Harras, K. A.," Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities", Future Generation Computer Systems, vol. 87, pp.278-289, 2018.

[3] Qiu, Y., Zhang, H., & Long, K, "Computation offloading and wireless resource management for healthcare monitoring in fog-computing-based internet of medical things", IEEE Internet of Things Journal, vol.8, no.21, pp.15875-15883, 2021.

[4] X. Wang and Y. Wu), "Fog-Assisted Internet of Medical Things for Smart Healthcare," in IEEE Transactions on Consumer Electronics, vol. 69, no. 3, pp. 391-399, 2023.

[5] Aazam, M., Zeadally, S., & Flushing, E. F.," Task offloading in edge computing for machine learning-based smart healthcare", Computer Networks, vol. 191, pp. 108019, 2021.

[6] Consul, P., Budhiraja, I., Arora, R., Garg, S., Choi, B. J., & Hossain, M. S," Federated reinforcement learning based task offloading approach for MEC-assisted WBAN-enabled IoMT,".Alexandria Engineering Journal, Vol. 86, pp. 56-66, 2024.

[7] Singh, J., Warraich, J., & Singh, P," A survey on load balancing techniques in fog computing," In 2021 International Conference on Computing Sciences (ICCS) pp. 47-52, IEEE Dec. 2021.

[8] Verma, P., Tiwari, R., Hong, W. C., Upadhyay, S., & Yeh, Y. H," FETCH: A deep learning-based fog computing and IoT integrated environment for healthcare monitoring and diagnosis", IEEE Access, Vol. 10, pp. 12548-12563, 2022.

[9] Asghar, A., Abbas, A., Khattak, H. A., & Khan, S. U.," Fog based architecture and load balancing methodology for health monitoring systems. IEEE Access, Vol. 9, pp. 96189-96200, 2021.

[10] Sheikh Sofla, M., Haghi Kashani, M., Mahdipour, E., & Faghih Mirzaee, R.," Towards effective offloading mechanisms in fog computing," Multimedia Tools and Applications, pp. 1-46, 2022.

[11] Singh, J., Singh, P.," A Sustainable Resource Allocation Techniques for Fog Computing. International conference on Sustainable Development Through Engineering Innovations," Lecture Notes in Civil Engineering, vol 113. Springer, Singapore, 2021

[12] Anand D., Kaur A., & Singh M., "Research on Internet of Medical Things: Systematic Review, Research Trends and challenges", Recent Advances in computer science and communications journal Volume 17, Issue 6, 2024, DOI:10.2174/ 0126662558248187231124052846

[13] Z. Wu, B. Li, Z. Fei, Z. Zheng, B. Li and Z. Han, "Energy-efficient robust computation offloading for fog-IoT systems," IEEE Trans. Veb Technol., vol. 69, no. 4, pp. 4417-4425, 2020.

[14] Li, Q., Zhao, J., Gong, Y., & Zhang, Q.," Energy-efficient computation offloading and resource allocation in fog computing for internet of everything", China Communications, vol. 16, no.3, pp. 32-41

[15] Z. Zhao et al., "On the design of computation offloading in fog Radio access networks," IEEE Trans. Veh. Technol., vol. 68, no. 7, pp.7136-7149, 2019.

[16] Ko, J., Choi, Y. J., & Paul, R., "Computation offloading technique for energy efficiency of smart devices", Journal of Cloud Computing, Vol. 10, no.1, pp 44, 2021.

[17] Chang, Z., Liu, L., Guo, X., & Sheng, Q., "Dynamic resource allocation and computation offloading for IoT fog computing system", IEEE Transactions on Industrial Informatics, Vol. 17 no. 5, pp.3348-3357.

[18] Daraghmi, Y.-A.; Daraghmi, E.Y.; Daraghma, R.; Fouchal, H.; Ayaida, M, "Edge–Fog–Cloud Computing Hierarchy for Improving Performance and Security of NB-IoT-Based Health Monitoring Systems", Volume 22, Issue 22. https://doi.org/10.3390/s22228646, 2022.

[19] B Premalatha, P Prakasam,,"Optimal Energy-efficient Resource Allocation and Fault Tolerance scheme for task offloading in IoT-FoG Computing Networks", Computer Networks, Volume 238, 110080, https://doi.org/10.1016/j, 2024 .

[20] H. Habibzadeh, K. Dinesh, O. Rajabi Shishvan, A. Boggio-Dandry, G. Sharma and T. Soyata, "A survey of healthcare Internet of things (HIOT) A clinical prospective," IEEE Internet of Things Journal, Vol 7, No. 1, PP 53-71, Jan2020.

[21] S, S.S.M., R, G., V, T.K. et al, "Adaptive heuristic edge assisted fog computing design for healthcare data optimization", Journal Cloud Comp (13, 127). https://doi.org/10.1186/s13677-024-00689-7 ,2024.

[22] Sufyan, F., & Banerjee, A, "Computation Offloading for Smart Devices in Fog-Cloud Queuing System", IETE Journal of Research,

RESEARCH ARTICLE

69(3), 1509–1521. https://doi.org/10.1080/03772063.2020.1870876, 2021.

[23] P. Kumar, H. S. Bhatia, A. Shrivastava, K. Yadav, M. Saraswat and D. Bisht, "Hybrid Metaheuristic Algorithms for Resource Allocation in Fog Computing Environments,"4th International Conference on Innovative Practices in Technology and Management (ICIPTM), Noida, India, pp. 1-6, https://doi: 10.1109/ICIPTM59628.2024.10563973, 2024.

[24] Kumari, N., Yadav, A., & Jana, P. K.),” Task offloading in fog computing: A survey of algorithms and optimization techniques”, Computer Networks, 214, 109137, 2022.

[25] David Lapp, Heart Disease Dataset (UCI Machine Learning). Kaggle https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset (Last accessed: March 2, 2024).

Authors

Dinesh Anand received Masters of Computer Applications and Master of Philosophy degrees from Guru Nanak Dev University, Amritsar, Punjab India, and Chaudhary Devi Lal University, Sirsa, Haryana, India in 2004 and 2007, respectively, He is pursuing Ph.D. degree from the Lovely Professional University, Phagwara, Punjab, India. He is also working as an Assistant Professor in Department of Computer Application, Guru Nanak Dev Engineering College, Ludhiana, Punjab since 2012. He has published several research articles in journals and conferences. He has more than eighteen years of extensive and diverse experience as a Researcher, Teacher, Member Board of studies, Course Coordinator etc. His research interests include IoMT, Cloud computing, Fog Computing.

Dr. Avinash Kaur is working as a Full Professor and HOD Cognitive Computing domain in the School of Computer Science and Engineering. She has published more than 60 papers in reputed journal, conferences, and book chapter. She has completed her PhD in Cloud computing and machine learning. Her area of interest is machine learning, and cloud/fog/edge computing.

Dr. Parminder Singh is working as a Full Professor and HOD Software Engineering domain in the School of Computer Science and Engineering. He has published more than 70 papers in reputed journals, conferences, and book chapter. He has completed his PhD in Cloud computing and machine learning. He has done a post-doctorate from UM6P, Morocco. His area of interest is machine learning, network security, and cloud/fog/edge computing.

How to cite this article:

Dinesh Anand, Avinash Kaur, Parminder Singh, “ Enhancing Healthcare Monitoring with Efficient Computation Offloading in Fog Computing ”, International Journal of Computer Networks and Applications (IJCNA), 11(4), PP: 506-518, 2024, DOI: 10.22247/ijcna/2024/32.