

Received December 18, 2021, accepted January 5, 2022, date of publication January 14, 2022, date of current version February 3, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3143793

FETCH: A Deep Learning-Based Fog Computing and IoT Integrated Environment for Healthcare Monitoring and Diagnosis

PARAG VERMA¹, RAJEEV TIWARI², (Senior Member, IEEE), WEI-CHIANG HONG³, (Senior Member, IEEE), SHUCHI UPADHYAY⁴, AND YI-HSUAN YEH³

¹Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab 140401, India

²Department of Systemics Cluster, University of Petroleum and Energy Studies, Dehradun, Uttarakhand 248007, India

³Department of Information Management, Asia Eastern University of Science and Technology, New Taipei 22046, Taiwan

⁴Department of Allied Health Sciences, School of Health Science and Technology, University of Petroleum and Energy Studies, Dehradun, Uttarakhand 248007, India

Corresponding author: Wei-Chiang Hong (samuelsonhong@gmail.com)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 110-2410-H-161-001.

ABSTRACT These days cloud-based infrastructure is facing many challenges, out of which the major issue is their syncing data before cutover and data migration. Due to the limited scalability in terms of security concerns of cloud computing, the need for a centralized IoTs based environment has been constrained to a limited extent. The sensitivity of device latency emerged during healthy systems such as health monitoring, etc. is the main reason, because healthy systems require computing operations on high-volume data. Fog computing provides an innovative solution to improve the performance of cloud computing, providing the ability to take the necessary resources and those that are closer to the end-users. Existing fog computing models retain several limitations, such as either considering result accuracy or overestimating response time, but managing both together impairs system compatibility. FETCH is a proposed framework that integrates with edge computing devices to work on deep learning technology and automated monitoring and offers a highly useful framework for real-life health care systems such as heart disease and more. The proposed Fog-enabled cloud computing framework uses FogBus, which demonstrates utility in the form of consumption of power, network bandwidth, jitter, latency, process execution time, and their accuracy as well.

INDEX TERMS Fog computing, edge computing, healthcare, machine learning (ML), deep learning (DL), Internet of Things (IoT), heart disease analysis.

I. INTRODUCTION

In the digital technology era, the use of intelligent devices in daily routine is experiencing tremendous growth in all sectors like medical, agriculture, etc. IoT-enabled devices use their sensing technology to generate big data and then transfer it via fog computing or cloud computing to destinations on which decision-making capabilities can be accurate by applying deep learning algorithms. However, fog computing, including cloud computing standards, has emerged as the backbone of a cutting-edge economy that uses the Internet to deliver services requested by users [1]. These two fields have taken significant consideration in the world of professions and scholars. But due to the heavy delay in response time, cloud computing is certainly not a decent option

The associate editor coordinating the review of this manuscript and approving it for publication was Mansoor Ahmed¹.

for applications that need a continuous feedback scenario. Advancements in technology including Big data handling with the Internet of Things (IoT), Fog computing, and Edge computing have grown important due to their nature as warm-heartedness and ability to deliver response properties that depend on the tracked target applications [2]. High volumetric storage capacity, computation, and the reliable communication practices provided by these emerging technology including edge gadgets, which encourage and improve mobility, safety concern, security, low scale latency, and bandwidth of the network so that fog computing seamlessly coordinates latency and continuous applications [2]–[7]. Currently, cloud computing structure similarly helps to generate the applications standards, for example, through IoT devices, fog computing, edge computing, and big data support and infrastructure [8], [9]. Routers, switches, computing nodes, and gateways are used by fog computing

to offer the least conceivable type of assistance with energy usage, network deactivation and the response time of the network.

Researcher K.C. Okafor [10] delivers a prediction algorithm based on dynamic reliability that has the ability to accurately estimate the average time to failure at the system level for the CloudMesh architecture. The researcher focuses on the QoS Proactive Auto Scaling Algorithm (PASCQA), designed with a heuristic approach for CloudMesh Cyber-Physical Systems (CPS). This proposed algorithm is capable of reducing scaling overheads, increasing the utilization of CPS resources, providing a highly flexible system for capturing user requests, and reducing physical connections in terms of size. But still the data delivery latency needs more attention in the context of high sensitive data like medical, etc. Also researcher Okafor *et al.* [11] works on optimizing bandwidth with edge devices. The researcher extended the fog and cloud spine-leaf architecture with the use of the Cisco Nexus platform to develop a scalable IoT datacenter including a cost-effective processing, storage and big data analysis environment.

Researcher Mutlag *et al.* [12] explores the difficulties of the use of fog computing in the application domain of medical care that has been investigated and distinguished that lethargy and response time are the most important and difficult to streamline quality of services (QoS). Medical services are one of the surest application areas that require accurate and ongoing results, and individuals have introduced fog computing in this field that promotes positive advancement in this technology. With the fog computing-based processing; we move assets to adjacent customers and subsequently reduce inactivity and expand the measure of wellbeing accordingly. Acquiring dizzying results suggests accelerated movements for core heart patents. In any case, the quickness of the results is not sufficient with such sensitive data that we cannot trade-off with the accuracy of the result. One approach to achieving high accuracy is by using the best in regular classroom exam programming that deep learning and their variance formulated on a vast dataset. Recently, deep learning [13] has seen excellent development in areas reaching from computer vision (CV) technology to the voice-based recognition system for the performance in general language preparation, system forecasting, and mixed-work information settings, all from discourse acknowledgment. In addition, ensemble technology-based learning [12] is used to fetch the best multiple classifiers among the number of classifiers as per the target requirement. Among this one of the ensemble, techniques are named bagging classifier, where the evaluator is on an arbitrary subset of the fit strains the base classifier information and later expresses his or her personal expectations on the ballot or, on average, to obtain a final forecast. Such evaluators help minimize the change when contrasted with a solitary evaluator by bringing randomness to the strategy of dataset dispersion. One other advancement of deep learning technology is forecasting and classifying the information from medical

services with surprisingly high accuracy [13]. As it may be, in the present scenario deep learning models for medical care applications are exceptionally sophisticated and required countless computational properties for both preparation and expectation [14]. Likewise, some investment is required to formulate this complex neural network and the data using for them must be isolated. Higher accuracy is required, the organization is more refined, and higher expected time [15], [16]. This has been an important issue for medical services and comparable IoT-based applications, where it is basic to get the results in a real-time scenario. As the computation at the edge has an exceptionally favorable state of decreasing response time, it provides other directions to the test of incorporating edge computing into complex deep learning models with the ultimate goal that we have achieved with high accuracy results in real-time scenarios. One of the central points of this work is to overcome the issue and suggest a tangible platform that not only results in low latency using the edge computing-based resources but in addition the use of the deep learning-based system to produce exceptionally precise results. These devices can give results of some work that has been done to perform a calculation on edge computing-based devices near the patient to reduce the resulting delivery time. A portion of this function actually depends on the simulations [4] and has not given a single competent system. In this way, the medical care industry is expected to work towards this deficiency.

The traditional signs of heart issues are difficult to distinguish and the patient needs an accomplished specialist to determine that a person has a heart issue. This is problematic to do for all intents and purposes due to the lack of experts as most nations do not really consider the computer-based applications or framework as an option to identify heart issues with the required accuracy and clarify capacity [17], [18]. Existing medical services frameworks, that are sent over the internet of things technology and pre-configured devices that are suitable for patient data processing to computing systems to the extent that results are communicated to customers within the cutoff time. Several earlier works have attempted to use IoTs to estimate the medical conditions identified with the heart but cannot ascertain the accuracy required by the tougher guidelines of clinical generalization agencies. In the recent past few years, later advancement as deep learning has gained fame and also outperform experts in the accuracy of the location of coronary disease [19], [20]. The work plans to unify deep learning in the medical services industry and IoT with the hope that it will adopt a model that offers low inactivity and high precision to reduce the issue of absence of specialists in clinical normalization promotes organization. There does not exist much such work, which means unifying these two parameters [21], although no one uses the idea of edge computing to improve precision using in-depth study models. We present a more thorough correlation in section 2 and subsection 7.9. Furthermore, the enhancement of the deep learning model is a specific extension to allow outcomes to be

ensured because it requires precise correction and a cautious balance of lethargy enhancement to offer the most desired support quality. Furthermore, expanding on previous works such as [2], [21], [22], FETCH (Fog Enabled Technique for Clinical Healthcare) gives a design for medical care computation that incorporates mixed backend structures such as FogBus [23] and Aneka [24], which makes it an optimal model.

Earlier works elaborate that there are two important types of medical care information classification schemes for cardiac patients using a variety of devices (for example, IoT-enabled sensors and recorded input data). The first short length of data is handled at the center of the fog node and the second is the high volume data prepared in the cloud data center (CDC) [1], [8]. Medical services receive quiet data by the organization at high speeds (configured as 250 MB for each moment or upgraded) [1]. Existing structures are not flexible enough to capture and result in two types of information situations and thus need to use edge and cloud computing enabled devices to take an application with such high volume data. After classification and collation of information from curious devices of IoT-enabled networks, information is stored away and handled by nervous nodes or cloud workers.

To deliver productive data administration to heart patients and various clients in need of ongoing results, a corporate edge to convey medical services and other haziness to delicate results with short response time, minimal energy usage, and high accuracy- Fog computing including cloud-based calculation model required. The absence of such models or structures that incorporate the force of high precision of the deep learning model, simultaneously with the low inactivity of the edge computing-based nodes prompted this work.

In the current work, we propose to FETCH: A Fog Enabled Technique for Clinical Healthcare system which is capable for in-depth study and automated diagnosis of heart disease including IoT and Deep Learning ensemble technique. FETCH gives medical care in the form of a mild mist administration, additionally, it is effectively related to the data of cardiac patients collated from various IoT-enabled devices. FETCH provides this kind of support by celebrative using of FogBus system [23] and demonstrates this application and designing to use the mist property to accomplish the same.

Some major contributions of this research paper are listed as;

- Propose a deep learning ensemble technique for the advancement of fog computing that supports a generic framework design
- Built-up a lightweight computing technique for heart patient diagnosis framework with the use of a deep learning ensemble technique named FETCH. This system is deployed with the utilization of the FogBus system to reconcile IoT-Edge-Cloud in a single unit for analysis the real-time data processing.
- The deployment of FETCH is investigating various performance measurements such as shown and accurate

execution, response time, rate of data transmission of the network, and the usage of energy.

The overall research is summarized into 8 sections, where section 2 presents the previous related work in the same direction. Research section 3 highlights the technical background of the work and its importance. The system architecture and front-end design of the proposed FETCH model are described in detail in the next sections 4 and 5. The various essential implementation modules of the model are briefly covered under Section 6. The next section 7 deals with the model performance evaluation part of the proposed model. The conclusion and future directions of the research work are included in the final section 8.

II. LITERATURE REVIEW

For the effective processing of healthcare data in the medical domain, fog computing provides an important paradigm, that can be fetched from various IoT-enabled equipment. The fog computing enabled devices or fog node for gauging gauze has the ability to deal with the heart patients' data, greatly reducing the latency, delay, or response time as edge computing enabled devices are much closer to the IoT enabled devices to compare to the cloud-based data centers.

An IoT-enabled medical services framework was proposed by the researcher He *et al.* [5]. The researcher introduced a board model called FogCapCare that incorporated a cloud layer with a sensor layer to detect patented heart health conditions and shorten the execution season for runtime readiness. FogCepCare uses a parceling and grouping approach and correspondence and similar handling strategy to advance exaction time. FogCepCare's presentation is in contrast and the current model uses a reproducible cloud climate and streamlines exaction time, although it falls short on performance evaluation as far as significant quality of services limitations, for example, power usage, inactivity, accuracy, and so on.

Another IoT-based e-health monitoring services framework is proposed by Researcher Ali & Ghazal [25]. This framework relies on the use of the software-defined network (SDN), which is capable of collecting information via cell phone as a voice control system, enabling the health status of patients.

An Autonomous-Monitoring-System (AMS) model proposed by the researcher Rajasekaran *et al.* [26] is particularly used for the Internet of Medical Things (IoMT). This model is capable to provide functionality for medical care. This research work is designed based on a reward mechanism that uses the Analytics-Hierarchy-Process (AHP) for the proper dispersion of the flow of energy between the considered nodes in the model. Cloud computing enabled simulated framework has been used to simulate and test the presence of the autonomous monitoring system as far as energy use and the autonomous monitoring system are better in caparison to the FGCS strategy, but the timing of the correspondence between nodes reflects the high latency of patient solicitation preparation.

A SMART-For-Gateway (SFG) model was proposed by researcher Constant *et al.* [27] to provide a potential filter, an intelligent filtering system, deep examination, and specific information exchanges during the IoT-enabled devices (wearable) and a system of data modeling in SFG. The proposed model streamlines the exposition about the execution of the time and the usage of energy in the system as well, although it doesn't think about the performance parameters of the system i.e. latency.

Azimi *et al.* [21] explore the efficiency of Convolutional Neural Network (CNN) based models in conjunction with classification techniques for research for Hierarchical-Edge-Based-Deep-Learning (HEDL) proposed for medical intensive as an illustration of a strategy for medical-care despite the IoT enabled framework. Rajkumar *et al.* [28] proposed a deep learning model with high scalability and accuracy for records the electronic health data of patients depends on Fast-Healthcare-Interoperability-Resources (FHIR) design. In the deep learning-enabled strategic environment the use of FHIR delineation in the proposed model provides the capability of accurately analyzing the multiple clinical opportunities from different focuses without harmonizing the site-explicit data. In addition, the proposed approach has been approved to use the clinical focus data in the form of electronic health records (EHR) of two US scholars, with 216,221 adult patients in the hospital at any rate 24 hours admitted and the accuracy of the expectation improves. Table 1 summarizes the comparison of the proposed mode FETCH with the available models.

An end-to-end scheme proposed by Moosavi *et al.* [22], to estimate the security for the IoT enabled devices under consideration of medical care, which uses the datagram transport layer security (DTLS) handshake protocol, which is separated without the need for reintegration at the physical layer of the devices- details established a secure correspondence between different spectacular routes. The proposed scheme results showed effective use in reducing communication overhead by 25% and latency below 16.00%. Expanding this work, our proposed work FETCH is expected to express an all-around incentive real framework and medical care application at centers of fog computing-based nodes.

A Framework proposed by researcher Rahmani *et al.*, [2] named Smart E-Health Gateway to use the required position of a gateway on the edge of a dedicated network to offer a variety of support, e.g., over time, with the real data mining process, data processing, data incorporation, and the local data storage. In addition, the proposed framework fully approves portable use to test IoT Enabled Warning Scores (EWS). Expanding this work, our proposed model FETCH highlighted the option of using an in-depth learning model in an ensemble technique to increase the expected accuracy and to prove more accurate results for native heart patients.

The challenges that analyzed in the previous works [5], [25]–[27], [29], [31], [37], [38] needs to address the full potential of IoT based computing technique for medical care framework;

- requires a productive IoT based healthcare applications, which can handle the information of a large number of cardiac patients with minimal energy use and short response time,
- An efficient estate planning method for processing conditions require that the client with the greatest asset use to execute the remaining burden to meet the cutoff time of the outstanding burden and
- The deep learning ensemble technique to continuously analyze the severity of coronary disease in patients.

III. TECHNOLOGICAL BACKGROUND

FogBus provides a framework that enables the development and the deployment as well, of the coordinated fog computing and cloud computing-based environment including the phase-free exaction of organized correspondences and uses [23], [39]. FogBus can connect with individual IoT-enabled devices that can be medical care sensors with the gateway devices to transmit the data and ventures to specialist nodes. The management of the resources and the initiated tasks are done at the fog representative nodes. To guarantee the data integrity, security, blockchain-based FogBus, user authentication, and encryption technology that enhance the unbreakable quality and the robustness nature of the fog environment. FogBus uses the HTTP RESTful API for correspondence and directs a continuous fog computing arrangement with the cloud computing by the use of the Aneka Software Platform [24].

Aneka provides a software platform that encourages the provisioning of applications that work on event and cloud computing [24]. Aneka gives developers an API to use virtual resources on the cloud. The center sections of the Aneka stage are planned and executed in a service-oriented fashion. A dynamic provision is the ability to powerfully accept assets and coordinate existing foundations and programming frameworks. In the best-known case, accepted resources are virtual machines (VMs) that are sourced from an infrastructure-as-a-service (IaaS) cloud supplier [40]. Aneka consists of dynamic provisioning as a component of fabric Services, which provides provisioning administration to deliver the virtual nodes from the public cloud suppliers to complement neighborhood assets. This is basically accomplished because of the relationship between the two administrations: Scheduling services and resource provisioning services. Currently, Aneka supports four diverse programming models [24], [40]: the bag of assignment models, the distributed strings model, the MapReduce model, and the parameter sweeping model.

At FETCH, we utilize the bag of assignment model for task appropriation in cloud-based virtual machines. FETCH utilizes the FogBus to bridge the cloud assets and Aneka to cloud-based resources.

IV. FETCH PROPOSED SYSTEM ARCHITECTURE

The FETCH model is a fog computing-enabled technology based on IoT ensemble cloud computing specifically

TABLE 1. A features comparison table of FETCH (proposed model) with the previous existing models and frameworks.

Proposed Models	Authored by	Features												
		Fog Computing	IoT	Deep Learning	Ensemble Learning	Heart Disease Prediction System	Power Consumption	Latency	Performance Parameters			Jitter	Training Accuracy	Testing Accuracy
									Execution Time	Arbitration Time	Network Bandwidth			
LCHM	Gia et al., 2017[29]	✓							✓					
FogCepCare	He et al., 2017[5]	✓							✓					
IoT e-health service	Ali & Ghazal, 2017 [25]		✓			✓								
ECGH	Akrivopoulos et al., 2017[30]	✓	✓											✓
GRAM	Choi et al., 2017[31]			✓		✓							✓	
SFG	Constant et al., 2017[27]	✓	✓				✓		✓					
HEDL	Azimi et al., 2018[21]	✓	✓	✓					✓				✓	
FIH	Mahmud et al., 2018[23]	✓	✓				✓	✓						
FogLearn	Barik et al., 2018[32]			✓										
CoSHE	Pham et al., 2018[33]		✓											
SLA-HBDA	Sahoo et al., 2018[34]								✓					✓
AMS	Rajasekaran et al., 2019[26]		✓				✓							
EOTC	Alam et al., 2019[35]	✓	✓											
CFBA	Abdelmoneem et al., 2019[36]	✓					✓	✓						
FETCH	Proposed work	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

designed for medical care services, which can adequately deal with the data of cardiac patients and analyze the situation well to distinguish the severity of the coronary disease. FETCH coordinates various instrumentation tools through the programming segments and permits organize to end the coherence of edge-fog-cloud for quick and accurate description of the results. Figure 1 presents the designs of the FETCH which consists of various tools and programming parts directly depicted.

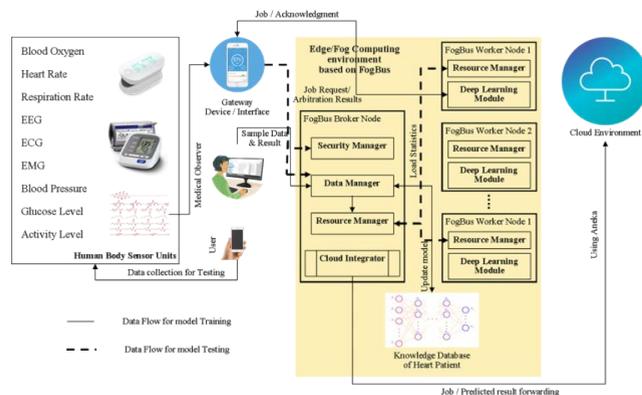


FIGURE 1. System architecture of FETCH.

A. HARDWARE COMPONENTS OF FETCH

The FETCH model includes the following device parts:

1) NETWORK ON BODY-AREA-SENSOR (BAS)

Three types of sensors establish this segment: medical sensors, actions sensors, and environmental sensors. This part provides the functionality to detect the data of the heart patient and carries it to the respective passage gadgets.

2) APPLICATION GATEWAY

The proposed framework used the three different types of devices for application gateway specifically consist for mobile phones, laptops, and tablets, which is going as fog enabled devices to collect the information from the different sensors and this information forwarded to the Brokers/Workers node for additional information.

B. FogBus ENABLING MODULE

A FogBus enabling module that includes

1) NODE FOR BROKER

This part receives the work demand or potential input information from a device of the gateway. The solicitation input

module receives the work demands from a gateway device that does not last long before information is transferred. The assertion module (part of the resource manager in the delegate node) takes all of the specialist nodes' stake management information and selects which node or sub-node to progressively send to the businesses.

2) NODE FOR WORKER

This segment especially works to complete the ventures assigned by the resource manager of the node of the broker. The node of the worker may have devices and single-board computers (SBCs) installed such as the Raspberry Pi processor. In the FETCH, worker nodes can be incorporate modern in-depth learning models to measure information and create and measure results.

3) CLOUD BASED DATA CENTER (CDC)

When the framework of fog computing becomes overloaded, due to that the arability of the services follows the latency or the size of the information is much larger than the normal size, at which point FETCH organizations harness the resources of the cloud-based data center.

C. SOFTWARE COMPONENTS OF FETCH MODEL

The programming leading parts of the FETCH model include;

1) PREPROCESSING AND REFINING THE COLLECTED DATA

The initial step after gathering the data from the input is the preprocessing to refine the data as per the expectation for the model. It involves the separation of information that uses information test evaluations. The discrete information is reduced to a more trivial measurement that is performed using set partitioning in hierarchical trees (SPIHT) algorithm, principal component analysis (PCA), and the encryption technique using singular value decomposition (SVD) to remove key segments of information vectors that affect the patients' health status.

2) MANAGEMENT OF THE RESOURCES

It consists of two modules, the remaining burden head, and the arbitration module [24]. The outstanding burden continues to solicit the tasks and undertaking to prepare instructional information. It additionally handles large chunks of information to be prepared. The interferences module resides in the node of the broker and selects whether the fog computing enabled node should be set to achieve the results, the actual broker, the fog expert node, or the cloud-based data centers [23]. The primary objective is to separate the work into different resources to accommodate the stack and give ideal execution. FETCH provides the control to the users to set the arbitration plans dependent on their own load balancing and applications requirements. The proposed scheme is depicted as the flow chart in figure 2 and the pseudo code is covered in Table 2.

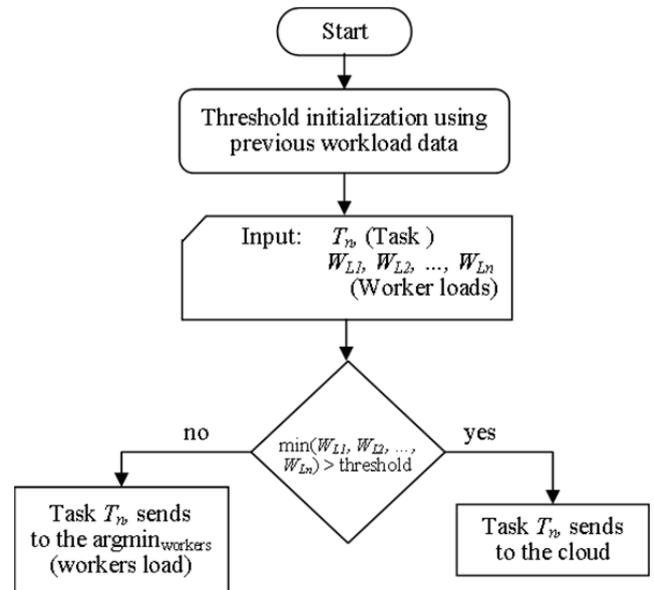


FIGURE 2. FETCH based resource scheduling.

TABLE 2. Pseudo code of FETCH based resource scheduling.

Set	Threshold T: sample data workload W over E (past data experience)
Input	T _n (working task), w _{L1} , w _{L2} , ..., w _{Ln} (Worker loads)
1	compute t _w = min(w _{L1} , w _{L2} , ..., w _{Ln})
2	if t _w > T, handover T _n to cloud (for further decision process)
3	else, handover T _n to arg min t _w (for further sample data computation)

3) DEEP LEARNING MODULE

Deep learning enabled model for the proposed system uses the data to design a neural network focusing on group information that includes vectors before handling information received from the body area sensor network. In light of the undertaking specified by the resource manager, it additionally predicts and creates results for the information received from the devices on the gateway.

4) ENSEMBLE MODULE

This module incorporated with the deep learning model to boosting the expected results from various models and uses a majority voting method to select the yield class that would be required if the patient had coronary disease. Similar to the weighted average, the majority vote base aggregates the outputs of the learners. However, instead of taking the average of the probability results, the majority voting base counts the learners' votes and predicts the final label as the label with the most votes. Compared to the weighted average, the majority vote is less biased towards a particular base learner's results because the number of majority votes outweighs the effect. However, event dominance in the ensemble model results from the favoring of a particular event by most of the same base learners or dependent base learners. The base learners

are the sum of weak learners (w_l) computed as of equation (1) [41];

$$S_L(\cdot) = \arg \max_k [\text{card}(l|w_l(\cdot) = k)] \quad (1)$$

This module resides in the FogBus hub which is returned to the assignment and is responsible for disseminating information and collecting results from other nodes of workers.

D. THE TOPOLOGY OF FETCH SYSTEM

All the major segments of the proposed FETCH system have been already described in the previous section including their required and processing amount of data, and control signals. In addition, correspondence should be considered under fault tolerance level and in persistent mode. This entire situation considered, with a selection of segments and the organization structure is presented in figure 1. Data transmission across all the devices on the edge is encouraged using the FogBus [23] and Aneka with the cloud-based virtual machine (CVM) [24].

Network topology covered under the FETCH system follows the hierarchy of master-slave style where the node of brokers acts as a worker to analyze and control the load. Under consideration of the non-cloud case, the device under gateway worker/broker sends work to the instance input information for the probe to run, at that time which runs the preprocessing, predication model and the computed results send back to the gateway devices. In the case of cloud forwarding, since the gateway devices cannot be on the virtual private network (VPN), it sends the information to the node of brokers and then brokers forward the collected data to the CDC. It likewise guarantees that the IoT sensors and the gateway devices are secure from the malicious elements and programmers as they cannot be connected to the web-based platform, but only local area networks (LANs) can be connected with the other fog nodes. Because the cloud-based environment is capable to access a large set of resources, the timing of the execution is dependent on both the broker's node and the CDC having higher correspondence overhead and less laziness due to the latency delay. At this point when the organization is empowered, the information received by the nodes of broker/worker is routed to any remaining edge and a large portion is picked up by the specialist node to which the information was sent through the bagging process (Slave node). FETCH system includes all the edge computing enabled devices that include the gateway devices, nodes of a broker, and working on a common LAN. The resource manager resides in the programming segment broker node and as a result, the gateway of broker job demands devices in a worker cloud manner that send work demands to it. The discretion result obtained from the resource manager is obtained by a gateway device that provides information about where to send the data. Three situations emerge here:

- Broker handling information to a node of worker
- Another node of worker for sending information
- Data processing based on the CDC (Cloud-based Data Center)

Depending on the situation, the gateway device may send information directly to the node of the worker or the broker (with or without the use of a cloud-enabled environment). The agent can allow the enumerated administration to send data only if it has enough devices and in addition, the worker node is overloaded. If the information has to be sent to the cloud, it passes through a broker node because the gateway cannot access the VPN that has a cloud-based virtual machine available. Additionally, workers sometimes send the broker a parcel of heartbeat to demonstrate that they are alive. The therapy data packet encapsulates a load of information in the same way that it is used by resource managers for load balancing.

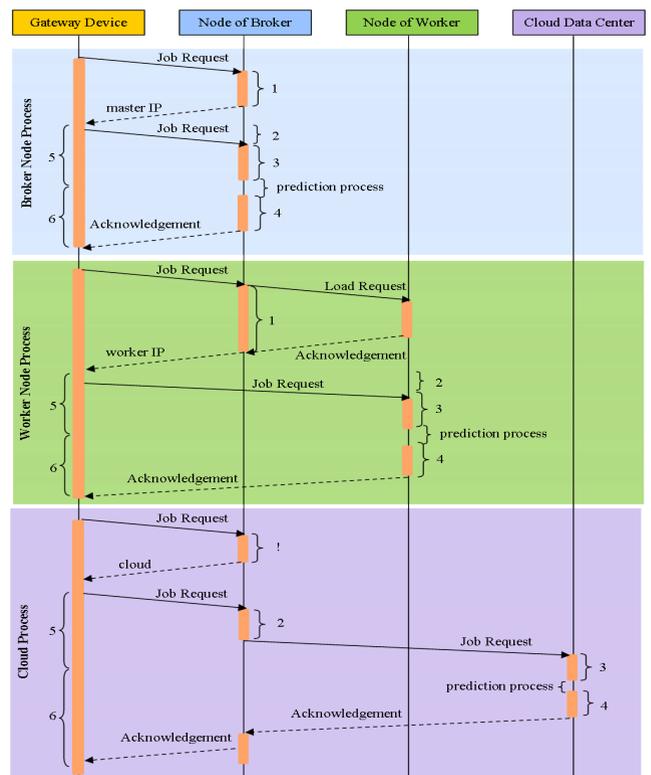


FIGURE 3. FETCH based communication sequence arrangements.

E. COMMUNICATION ARRANGEMENTS

In the FETCH system, all the hardware equipment depends on the predefined convections as depicted in figure 3 for three different conditions i.e. Brokers node, Worker node, and Cloud Center that have been described in the previous section. In each case, the gateway initially sends a job solicitation to the broker node. Depending on the situation, the broker sends either the worker IP address (of a similar LAN) or the master IP address (with or without a cloud-based environment) to the gateway nodes. In the case of a broker, the broker node can check the worker's load. On the off chance that all workers have a substantial load or are all low and the cloud has been erased, the broker at that time sends the gateway device without sending its IP to the cloud. If the workers there are not stacked as fast, the broker at the point sends the IP

address of the most un-stacked worker hub to the gateway device. Expanding the number of workers will increase the time of discretion as the investigation of overloaded should end. Under consideration of the non-cloud case, the gateway device sends a task to the nodes of broker/worker for example input information for the exam, which at that time runs the preprocessing, prediction model, and sends back the computed results to the gateway devices. In the case of cloud sending, since the gateway device cannot be on the VPN, it sends information to the broker node which then sends it to the CDC. Likewise, it guarantees that the IoT sensors and passage gadgets are separated from hazardous materials and programmers as they may not yet be connected with the internet and have LANs with other fog computing-based nodes. Because the cloud-based environment is capable to access a large set of resources, the timing of the execution is dependent on both the broker's node and the CDC having higher correspondence overhead and less laziness due to the latency delay. At this point when the ensemble is empowered, the information received by the worker/ broker node is routed to any remaining edge and the greater part is picked up by the worker node, using the data sent through the bagging technique.

V. PROPOSED DESIGN OF FETCH SYSTEM

The model of fog computing that is already depicted in the previous section considered the data of heart patients as a contribution from the sensor and the computed results send back to the data center in which the patient has the coronary disease, with case certainty. It is executed with parts that include information of preprocessing modules, ensemble deep learning modules, and the interface for gateway depicted directly.

A. DATA PREPROCESSING PHASE OF HEART PATIENT

Information from basic pulse-oximeter or the devices of ECG is in a plain graphical organization and one must be prepared in advance to find the estimates for several highlights of contributions to deep learning models [42], [43]. The application-specific domain information is required to be taken care of in the framework. Normalizing age information as it was demonstrated in figure 4. Essentially, the data of rest blood pressure is similarly slow and patients with coronary disease indicated more severe hypertension with patients not having the coronary disease. In the same way that understanding cholesterol levels reflects some objective clear behavior, good health patient's communication is leptokurtic. In fact, even with the most extreme pulse, all patients who are considered under healthy status have significantly high pulse rate i.e. approximately 160 in comparison with those patients having heart disease i.e. approximately 150. Various highlights such as chest pain and fasting glucose should be constantly changed from virtue to absolute virtue. Similarly, the peak exercise ST section and the cardiac position recovered from the thallium test.

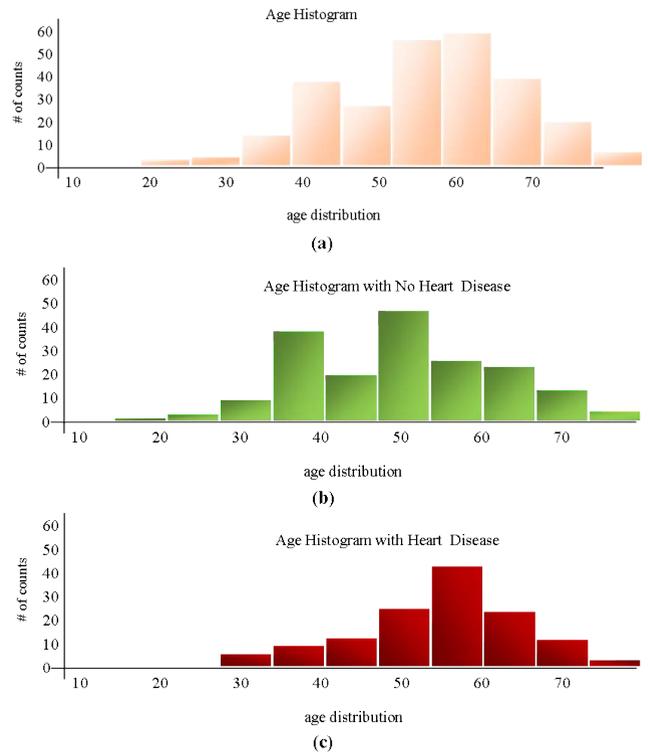


FIGURE 4. Age distribution of Heart patients in the Database including, (a) all considered patients in the dataset, (b) patients having no heart disease, and (c) patients having heart disease.

B. APPLICATION OF DEEP LEARNING ENSEMBLE TECHNIQUE

A deep neural network (DNN) ensemble technique has been used in this model for predictive analysis and our thoughtful application. The model is used to solve the purpose of binary classification problems. The first model was formulated on the patient's heart continuous data from the Cleveland dataset with known output classes and then the corresponding training model is used to estimate the results of the continuous information contribution as shown in Figure 5.

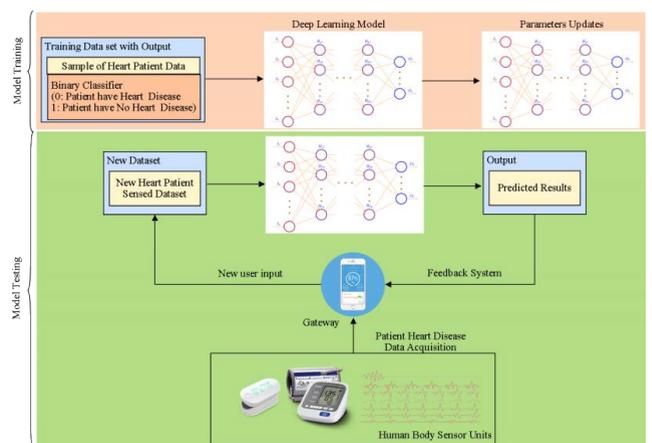


FIGURE 5. FETCH model training and testing phase.

The dataset is split into the ratio 7:2:1 with training, testing, and validation respectively. The 70% ratio of the dataset is used as the training dataset used to prepare the model, the 10% ratio of the dataset used as validation dataset that is used to set the model, and the 20% ratio of the dataset used as the testing dataset that is used to test how the model performs on new information. The training model can be placed in all nodes previously fitted for handling by placing them away in a specific dataset. Another methodology may be used to train the model independently, which focuses on datasets drawn in a different model. In preparation for propagation, information dissemination promotes processes, which test hastily information from the dataset with replacement and send it to different edge hubs to produce a singular model [44]. During the time of diagnosis, at any point a hub has resumed an undertone; it receives patient information that is a vector of shape. This information is taken into account t as a contribution to the model, making a forward pass on to the DNN, for the binary output i.e. either 1 or 0 that means either the patient has a coronary disease or not respectively. At the time of diagnosis, we are using the bagging process-an an ensemble technique to join the result of different models to produce results with high accuracy. The worker gets the information, which is handed over to other worker nodes. At that point, every worker adds it to their line and the predicted results of every node of worker are sent back to the worker allowed for this assigned task. At that time the major part expectation class is sent from bagging to the devices of the gateway. FETCH model provides the functionality to the user to cripple this element when the required results are inertia root. In the resulting section 7, we showed the performance of the deep learning model with ensemble technique that leads to better improvement and high response time with the network overheads.

C. COMMUNICATION MODE WITH ANDROID INTERFACE

FastHeartTest and android executable interface have been used to send the data to the broker or worker nodes in the gateway devices. The appearance of the application interface is shown in figure 6. This appearance allows the gateway to act as an intermediary between the body sensor network and the node that works as a worker. Correspondence is accomplished using the HTTP RESTful APIs. We used the HTTP Post method to try and download the result information from the gateway devices. Every broker node, worker node, and the CDC include a pre-training exercise handily by the deep learning model and preprocessing programming model.

VI. MODEL IMPLEMENTATION MODULE

All the parts referenced in section 5 were executed in different programming languages. The preprocessing and the deep learning with ensemble technique segments of the implementation were realigned to use a python programming environment. The preprocessing module standardizes the information dependent on the min and max estimated value of the area boundaries in datasets and their spread.

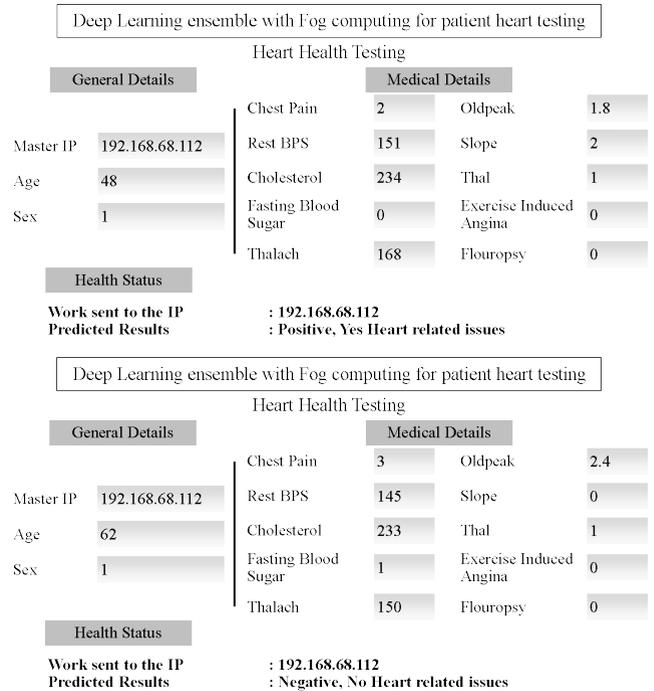


FIGURE 6. FETCH model Gateway of a user interface for both negative and positive patient heart-related issues.

TABLE 3. List of model parameters.

Input layer Size	13	# features of the considered dataset
Output layer Size	2	# 0 or 1 i.e. binary classifier
#hidden layer	3	FC: Fully connected layer including 20 nodes, 20 nodes and 10 nodes respectively
Model Optimizer's	"Adam"	
Model Activation function	"ReLU"	
Learning rate of the model	0.0001	

The deep learning with ensemble technique application enabled using the scikit learn – Machine learning Library in python environment [45]. Bagging classifier a method of scikit learn library has been used to implement the voting scheme in our proposed model. The model base takes the basis of the classifier which is a deep learning network for the considered situation and the number of classifiers used as input. Currently, the model haphazardly transmits the information among the classifiers to prepare the model. In the time it accepts all the expected classes as input and the predicted output of the major parts. Table 3 covers the parameters of the best base model on the considered dataset after tuning to the next:

The application of the Android interface was built using MIT's app inventor that establishes a communication link with the FogBus node of the broker. The Android interface holds the information credit formatted documents in comma-separated values (.csv) and transfers it to a broker node in the Data Catalog module using HTTP POST.

Arbitration modules are assigned to the node of brokers additionally that selects which worker node to perform the task. This selection cycle of the worker node is performed according to the default FogBus strategy of selecting the workers with the least CPU load. Any worker node which is selected, a comma-separated value formatted record is sent for examination. The execution interface module in each expert receives information and initiates the deep learning ensemble technique code to check the information. The returned results are sent back to the worker/broker node that sent the information documents. The results are ensured to use the bagging technique and sent to the gateway decides.

A depiction of the various moduli and their collaboration is shown in figure 7.

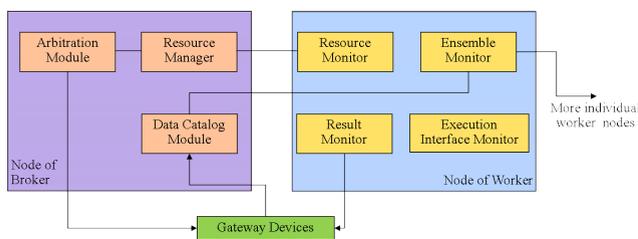


FIGURE 7. Various modules of the FETCH model.

VII. MODEL PREFORMATION EVALUATION

The proposed FETCH model is demonstrated to check its availability; feasibility and efficiency with executed, communicated, and deployed the model on an actual Fog enabled system of devices using the FogBus framework [23]. The model has been used for a true use of identifying heart issues for patients in the use of state-of-the-art deep learning strategies using an environment of fog-based learning. We have dissected the precision and the response time with the network and the energy overhead shows that the proposed model is profitable and has low overheads.

A. AN EXPERIMENTAL SETUP DETAILS

The experimental setups for evaluation of the proposed system FETCH and configuration of the tools are depicted as follows;

- **Gateway Devices:** LETV S2 4G enabled with Android 10.
- **Nodes of Broker (Master node):** Lenovo Yoga ThinkPad X2 with Intel i7-250 GHz processing unit, including 16GB RAM and 240 SSD ROM. The machine was configured with 64-bit professional windows 10. For the deployment toolkit, Apache HTTP server 2.4.34 has been used successfully.
- **Worker (slave node) nodes:** Raspbian stretch Operating System with Raspberry Pi 3B+ (μ P), ARM Cortex SoC processing unit with 1.4GHz, including 1GB LP-DDR2 SDRAM and a wifi unit (IEEE 802.11) with deployment toolkit apache HTTP server 2.4.34 have been used.

- **Public Based Cloud:** Microsoft Azure B1s Machine including 1v-CPU, 1GB-RAM, 2GB-SSD, Windows Server 2016.



FIGURE 8. The working environment of the FETCH model.

Figure 8 shows the actual performance of the proposed system model. During the analysis process, specific data parameters are considered and the data was recorded using the Microsoft Performance Monitor on the master and Azure enabled on virtual machines, although NMON performance monitors are used in Raspberry Pi IC (integrated circuit) (Source: Microsoft Windows performance toolkit. <https://docs.microsoft.com/en-us/windows-hardware/test/wpt/>, SplunkBase performance monitor. <https://splunkbase.splunk.com/app/1753/>). Microsoft Network Monitor 3.4 was used in the broker's (Source: Microsoft Network Monitor 3.4. <https://www.microsoft.com/en-au/download/details.aspx?id=4865>) nodes and Raspberry Pi in VnStat (Source: vnStat Network monitoring tool. <https://humdi.net/vnstat/>) was used to use organization transmission capability.

B. DATASET DESCRIPTION

To perform the experiments for computing the test results, we considered the information from cardiac patients to search for the presence of coronary disease in the patient [42], [43], [46], [47], having a binary-valued 1 (presence of heart disease) or 0 (no presence). The dataset Cleveland (Source: <https://data.world/uci/heart-disease>) [46] is used to lead the analysis that was formed by Andras Janosi, Doctor of Medicine at the Gottsegen Hungarian Institute of Cardiology, Hungary, and others. Patient's names and their patient numbers are classified. 14 relatively important details have been used in this research work for the classification of the patients. These considered 14 relatively details about the status of the patient have been detailed in Table 4 and exemplified in Table 5.

C. EXPERIMENT SYSTEM CHARACTERISTICS

The dataset described in the previous section is used here, testing the model of whether a patient has a coronary disease or not, and based on an estimate of the threshold set for each patient. For complete information, the dataset was divided into two-three different datasets, namely training, validation,

TABLE 4. Dataset description of 10 heart patients.

S. No.	Parameter	Description
1	Age	in years
2	Sex	Two values (1=male; 0=Female)
3	Cp	Chest pain type
		1. Typical Angina
		2. Non Typical Angina
		3. Non Angina Pain
4. Asymptomatic		
4	trestbps	Resting blood pressure (in mm Hg on admission to the hospital)
5	chol	Serum cholesterol in mg/dl
6	fbs	Fasting blood sugar > 120 mg/dl
		1 = True; 0 = False
7	restecg	Resting electro cardio graphic results
		0: normal
		1: having ST-T wave abnormality (T wave inversion and/or ST elevation or depression of > 0.05 mV)
		2: showing probable or definite left ventricular hypertrophy b Estes' criteria
8	thalach	Maximum heart rate achieved
9	exang	Exercise include angina (1 = Yes; 0 = No)
10	oldpeak	St depression induced by exercise relative to rest
11	slope	The slope of the peak exercise ST segment
		1. up sloping; 2. flat; 3. Down sloping
12	ca	Number of major vessels (0-3) colored by flourosopy
13	thal	3 = normal; 6 = fixed defect; 7 = reversible defect
14	Target (num)	Diagnosis of heart disease (angiographic disease status)
		0. <50% diameter narrowing
		1. >50% diameter narrowing
		(in any major vessel)

TABLE 5. Sample of Cleveland dataset of with binary target values of heart disease patient.

s.no	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
1	67	1	0	160	286	0	0	108	1	1.5	1	3	2	0
2	67	1	0	120	229	0	0	129	1	2.6	1	2	3	0
3	62	0	0	140	268	0	0	160	0	3.6	0	2	2	0
4	63	1	0	130	254	0	0	147	0	1.4	1	1	3	0
5	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
6	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
7	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
8	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
9	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
10	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

and testing with ratios of 70%, 10%, and 20%, respectively. The initial section was used to prepare the model, the second for approval, and to tweak the model parameters. The previous part was used to test the model exposition. To determine the amount of detail of the FETCH model, this merit was noted.

The accuracy of the model prediction: 1807 models consist of the dataset, of which 1355 were used to train the models

and 452 were uses to testing the accuracy of the model. The testing of the model was divided equally among all the nodes of broker/worker to acquire their specially tailored deep learning model. All hubs will need to be transmitted to use all assets to produce a dataset model as a quantity of fog computing nodes. This reduces the preparation time plus the accuracy of the test. To notice such effects, the accuracy of the preparation and testing was investigated. We characterize absolute as the level of all official absolute patients, for which the model effectively predicts if they have coronary disease. We think of improvement of various blur settings, by changing the number of edge nodes and with or without ensuring the results.

Characteristics of time: Subsets that represent the specific timing thresholds that appeared in figure 3 were additionally observed and considered. These include interference time, latency, execution time, and jitter. We are comparing these timing parameters for various fog computing settings that have no node of edge computing or up to 2 nodes of edges (with or without ensemble) or a cloud having a calculation framework.

Range of bandwidth usage by the Network: For example as a node of broker, worker, or cloud and the number of worker nodes influence the network usage that was focused on exploring the use of the network in various cases. Like the analysis of deadlines, we analyze the network transfer speed usage for diverse fog computing scenarios. It was never really out of the dependence of data transmission used with the various fog computing configurations that FETCH provides.

The system power consumption: Energy is an important motive behind moving from cloudy to fog domain, in addition to this we also considered the use of force in various situations. In light of the force, use is considered and before drawing separate tests we examine how a unique FETCH design can be used for different users and application prerequisites.

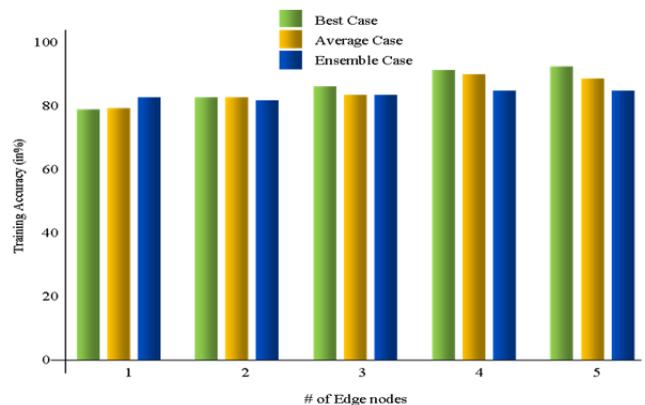


FIGURE 9. Variation in training actuary with number of edge nodes.

D. ACCURACY OF THE PREDICTED MODEL

Variation in the accuracy of the training module with the different nodes of edge computing (brokers nodes other than

worker nodes) is shown in figure 9. We can see that the accuracy of the training steadily increases as the number of workers increases. It is on this basis that each node learns a model for the information it receives, and as the number of nodes increases, the number of models obtained by each node decreases, and therefore the model for different epochs over fitted the samples and hence the accuracy of the training increases.

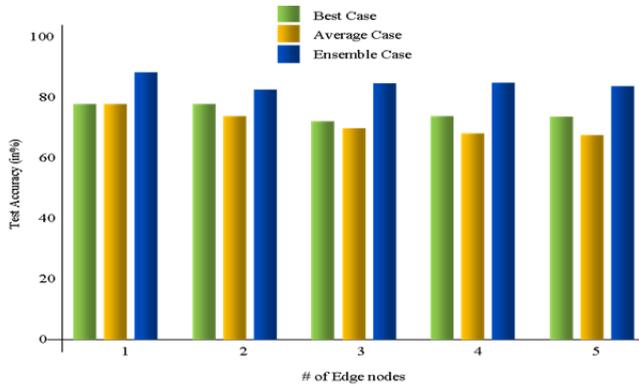


FIGURE 10. Variation in test accuracy with number of edge nodes.

In figure 10 we show the variation of test information precision as the quantity of edge computing nodes increments. As per the expectation, the accuracy of the test decreased with a greater number of nodes because each node gets a more modest subset of the training dataset and therefore it cannot sum to the model. Another assumption is that adapting ensemble learning gives consistently better performance without the ensemble learning-best or average case.

E. CONFIDENCE LEVEL FOR PREDICTION MODEL

At whatever point the prediction model with deep learning techniques is enabled, if the patient has coronary disease, it creates two possibilities P_0 : no heart disease probability and P_1 : heart disease probability, an ultimate disease that $P_0 + P_1 = 1$ with. An expectation (P_0, P_1) is measured as 100 (2 max P (P_0, P_1) 1) and thus has a range of [0, 100]. In this way, on the prediction that the probability is (0.5, 0.5) the certainty at the confidence level is 0 and when they are (0.9, 0.1) the predictor class is 0% and the level of certainty is 80%. The confidence variance shown in figure 11 represents the binary classifier for the total test dataset, the subset on which the model made an accurate estimate, and where the prediction was inaccurate.

We see that the certainty is higher for information where the focus was correct, where predictability was in contrast to information where the expectation was false. The greatest certainty with which the model predicts 49.7% by mistake is likely to be closed in such a way that the certainty is less than 50%, at which time our model advises the patient’s consult because the prognosis can be problematic.

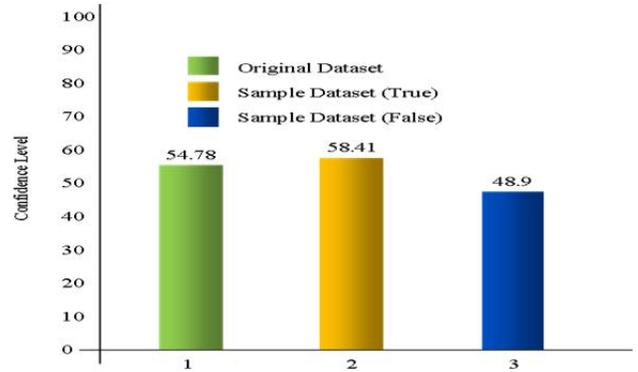


FIGURE 11. Confidence level of the model with a different set of samples of the dataset.

F. ANALYSIS OF THE TIMING CHARACTERISTIC OF THE MODEL

Variation in the interference time of the model broker node for different fog calculations conditions:

- Specifically for the broker,
- Individual worker nodes,
- Two worker nodes, and
- The cloud environment is shown in figure 12.

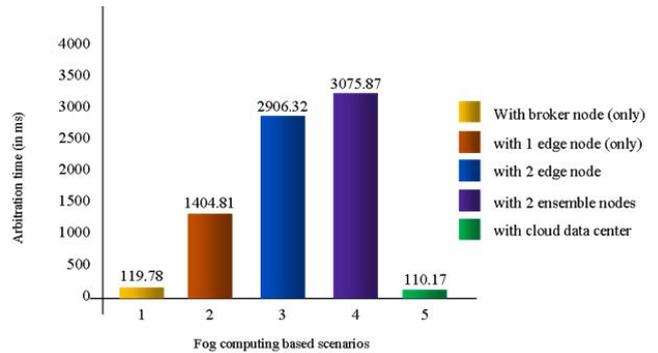


FIGURE 12. Results of arbitration time (in milliseconds) with different fog computing scenarios.

We have seen that when the assignment is to be sent directly to the broker or master’s node or to interfere with the cloud the time is almost negligible which is approximately being 115ms. As the amount of edge computing nodes increments, the broker nodes need to check the load on each worker node and find out the base load of the individual working person, later interference time increases because of edge computing. The amount of nodes increases. At this point when the information is passed to labor nodes to assemble learning, the broker does not need to examine any stack at that point, as the decision for most classes is terminated by one of the worker nodes, Therefore, arbitration is like time without ensuring the case.

The variation of the latency shows in figure 13, which is according to figure 3, is the extension of the corresponding time and the latency delay. We can observe that if the task is sent to the broker or any of the edge computing nodes, the inactivity at that point is approximately equal to all correspondence that occurs via single-node data transfers.

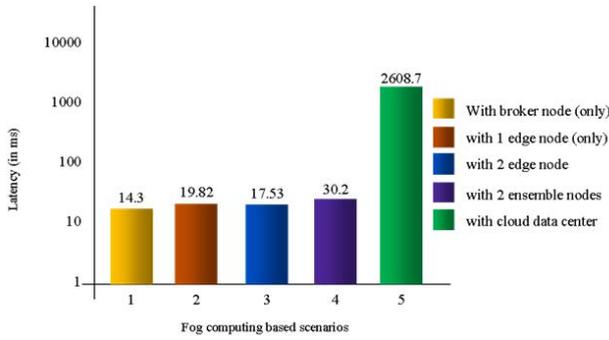


FIGURE 13. Results of Latency (in milliseconds) with different fog computing scenarios.

Among the ensemble cast, the value of latency is a bit higher. Under considering the cloud-based system, latency is more due to the multi-node transmission of the data outside the LAN.

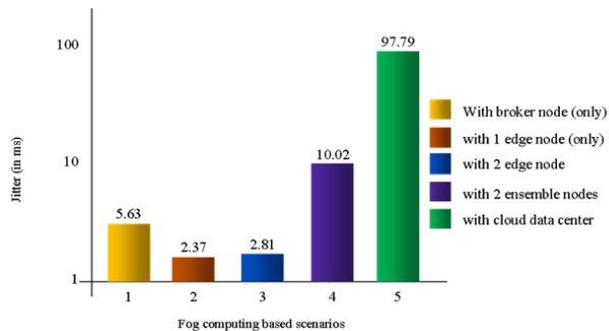


FIGURE 14. Results of Jitter (in milliseconds) with different fog computing scenarios.

The jitter value represents the variation of response time for consecutive task demands. This is a basic parameter for most continuous applications, including test information. Figure 14 guided the variation of jitter with the configuration of fog computing. After observing the results we identified that the jitter is higher for the broker than the case in which the undertaking sends data to labor nodes. It is on this basis that discretion in various functions, management of resources, and security checks are additionally carried out by the broker. As the number of worker nodes increases, there is some increase for the two edge computing nodes as opposed to single edge computing nodes due to differences in the heap of workers. Jitter is additionally high in the condition of the considered ensemble. When the CDR is turned off, the jitter response is very high.

The variation in the execution time has been shown in figure 15. As per the expectation, the execution time is shorter in considered cloud systems due to higher asset access. Representative execution time is shorter than expect execution because workers in het FETCH system are Raspberry Pi enabled with fewer clock repetition processors. Similarly, when the prediction used in the ensemble technique is stronger than the time of execution, the worker node now needs to test which class is larger among all the considered predicted classes.

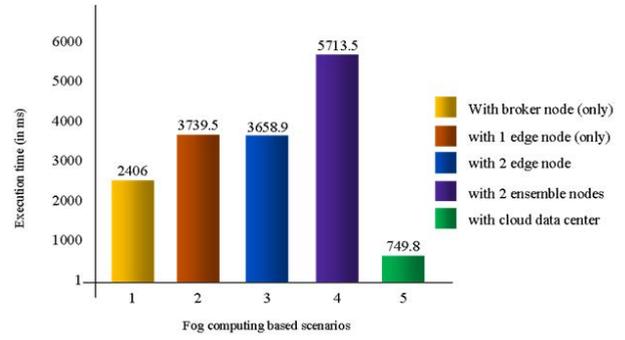


FIGURE 15. Results of Execution time (in milliseconds) with different fog computing scenarios.

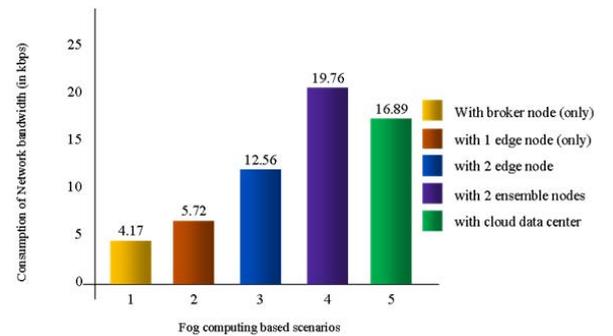


FIGURE 16. Results of consumption of network bandwidth (in kbps) with different fog computing scenarios.

G. CHARACTERISTICS OF THE NETWORK BANDWIDTH USAGE

The variations of network bandwidth usage of all the considered edge computing nodes under different conditions are shown in figure 16. We see that increasing the number of worker nodes requires heartbeat data packets, similar to the network usage expert. Security checks and data transmission tricks (including cloud) are highly essential. The most notable case when ensemble technique is considered is the use of network transmission capability in gathering information, as it is sent to all workers' nodes.

H. THE SIGN OF POWER OF THE SYSTEM

In addition, we tried that the FETCH system is using energy consumption properties in various conditions. The CDC's force use is in high contrast with the Broker node (as the laptop in our considered setup) or the worker node (as Raspberry Pi microprocessor in our considered case). This indicates the use of exceptionally high forces in the cloud case as opposed to the edge computing case. As worker node volume increases, the consumption of the power used by the FETCH system increases (shown in figure 17).

I. WORK ANALYSIS WITH THE EXISTED WORK

The various works propose to compute the models for the application of medical care using the fog computing technique but they don't consider the different approaches that the FETCH system does. Much earlier work [28], [31], [42]

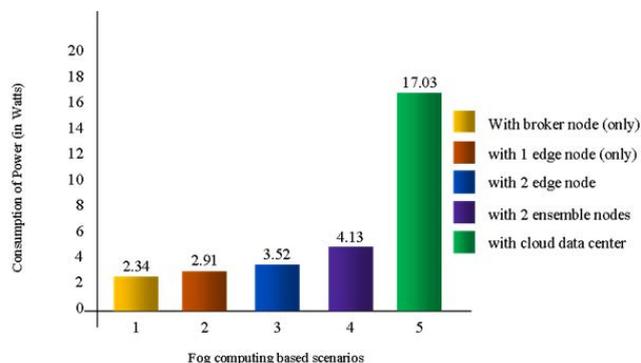


FIGURE 17. Results of consumption of power in watts (W) with different fog computing scenarios.

does not use the influence resources that are closer to the edge of the considered network. According to figure 13, such models give very high latency because all the calculations are performed on the cloud and consequently transfer high data. Including the advanced technology i.e. artificial intelligence-enabled deep learning algorithm-based prediction model, the proposed model FETCH can make the best use in the class neural network models for exceptionally accurate prediction of health characteristics of patients. Various other works such as [2], [9] or more related works done by other researchers like [5], [27], [28], [47] do not have the ability to include such models and deliver lower disease detection system with accuracy. It is decisive in basic medical care applications, particularly for people concerned about heart-related issues such as heart attack, heart stokes, or arrhythmia disease to deliver low sleepiness and deeply accurate results. In addition, deep learning that uses in the previous works by [23], [42], [45] utilizes the ensemble technique for similar computation and fundamentally high accuracy, using the strategies that ensure surprisingly superior results. As mentioned in the demonstration section 7.4, with the ensemble technique, the accuracy rate of the prediction is increased by 17% with the considered case including 5 edge nodes, which cannot use exiting framework i.e. deep learning with ensemble technique provided.

In addition, the dissatisfaction with the earlier work, the FETCH utilizes the FogBus system [25] to present a hybrid design with various accuracy rates, response times, network characteristics, and the consumption of power characteristics as well. In light of different applications and user requirements, the use of different setups can be illustrated in the accompanying sections. This allows the user to modify the system structure according to their requirements. This non-insignificant increase of cohesion and synchronization between the nodes of fog computing allows the execution of deep learning using the ensemble technique that improves the transition search precision as well as the energy according to the various requirements. Thus, the FETCH system gives the improved architecture for medical representatives of healthcare computation that was not proposed by any existing work.

J. RESULT DISCUSSION

In the prior work [25], examination with the strength of FogBus and earlier such fog computing enabled systems demonstrated how FogBus gives more efficient execution of the use of dealing with edge and cloud assets. This work utilized the FogBus system with simplistically designing the latency and the precise sensitive use of the heart patients and quiet examination and provided an ideal opportunity to use the resources of edge and cloud effectively. The application development system gave various setups that provide better accuracy or inactivity depending on the user requirements. Given exploratory results, we propose the FETCH system to be used as an on-target application configuration based;

For latency-critical and the targeting tasks that are considered lightweight or the energy bounded conditions, the considered worker nodes should be used to fulfill the requirement. This gives an exceptionally short result time due to the proximity of the worker nodes. On the off chance that the energy and the network data transfer capacity or network bandwidth constraints requirements are in place, then the bagging - ensemble technique should be deactivated but in the case, if it is not deactivated it will enabling the bagging that will support the better accuracy.

A CDC configuration with adequate and latency open views should be used in any case such that the task would not have the option of effectively completing on the resource constraint edge of worker nodes.

The proposed model is limited and dependent on the past workload data values, which supports boosting the nature of decision-making over minimizing the error. Model reliability is high over the low features noise and bias values.

VIII. CONCLUSION AND THE FUTURE DIRECTIONS

Medical care as an assistant is a vast and sensitive project. In the current research work, our main objective is to focus on the healthcare system for heart patients, which propose a Fog computing enabled smart healthcare system that incorporated with the latest technology i.e. deep learning ensemble technique that provides an automated diagnosis of heart disease with the utilizing of FETCH system for IoT enabled resources. FETCH delivers medical care in the form of fog computing services and effectively manages the data of a heart patient user that come from various IoT-enabled devices. FETCH incorporates the deep learning technique in edge computing enabled devices and deployed these devices in the environment for real-life use of heart disease testing. In the prior art for such a heart patient examination did not utilize the deep learning technique and resulted in an exceptionally low accuracy rate in the prediction power, which rendered them redundant in practical settings. The AI-based deep learning model with high accuracy requires exceptionally high processing resources (that includes CPUs or GPUs) for both purposes i.e. to train the model or predicted the model. This work allowed the network with a complex deep learning model to set in edge computing standards that

use the correspondence and model dispersion methods, such as to ensure that high latency can be met with low latency. It was approved for examining real-life heart patient data through formulating neural networks on well-known datasets and sending a working framework, which provides real-time prediction capability. We have used the FogBus structure to approve the FETCH in an environment of fog computing and try the effectiveness of the proposed framework regarding consumption of power, network bandwidth, latency, jitter, the accuracy of the training model, testing precision, and the time of execution.

As per the future direction of this proposed work, the extension work of FETCH allows for cost-optimal exaction to provide a specific quality of service features and a hybrid fog-cloud cost model. In the present work, FETCH works with the record-based dataset that can be continuously incorporated to capture the data directly from the sensor to make it easier to understand. In addition, the model preparation system currently uses a different preparation on each worker node in the system. The models formulated at each node are consolidated using various aggregated models of stowing. More curiosity models may be additionally expressed to improve accuracy. In addition, the proposed architecture can be made robust and nonexclusive to support other fog computing-based applications, for example, smart production lines in the agricultural sector, weather forecasting domains, smart city forecasting domains, and traffic management for industry domains. FETCH can likewise be expanded to other important areas of healthcare, for example, cancer, diabetes, and hepatitis, which can support efficient healthcare services to the concerned user or patient.

REFERENCES

- [1] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.
- [2] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, "Exploiting smart e-health gateways at the edge of healthcare Internet-of-Things: A fog computing approach," *Future Gener. Comput. Syst.*, vol. 78, pp. 641–658, Jan. 2018.
- [3] S. Tuli, N. Basumatary, and R. Buyya, "EdgeLens: Deep learning based object detection in integrated IoT, fog and cloud computing environments," in *Proc. 4th Int. Conf. Inf. Syst. Comput. Netw. (ISCON)*, Nov. 2019, pp. 496–502.
- [4] S. S. Gill, R. C. Arya, G. S. Wander, and R. Buyya, "Fog-based smart healthcare as a big data and cloud service for heart patients using IoT," in *Proc. Int. Conf. Intell. Data Commun. Technol. Internet Things*, 2018, pp. 1376–1383.
- [5] S. He, B. Cheng, H. Wang, Y. Huang, and J. Chen, "Proactive personalized services through fog-cloud computing in large-scale IoT-based healthcare application," *China Commun.*, vol. 14, no. 11, pp. 1–16, 2017.
- [6] I. Abdullahi, S. Arif, and S. Hassan, "Ubiquitous shift with information centric network caching using fog computing," in *Computational Intelligence in Information Systems*. Cham, Switzerland: Springer, 2015, pp. 327–335.
- [7] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.
- [8] A. Goyal, K. Narang, G. Ahluwalia, P. M. Sohal, B. Singh, S. T. Chhabra, N. Aslam, B. Mohan, and G. S. Wander, "Seasonal variation in 24 h blood pressure profile in healthy adults—A prospective observational study," *J. Hum. Hypertension*, vol. 33, no. 8, pp. 626–633, Aug. 2019.
- [9] S. S. Gill, P. Garraghan, and R. Buyya, "ROUTER: Fog enabled cloud based intelligent resource management approach for smart home IoT devices," *J. Syst. Softw.*, vol. 154, pp. 125–138, Aug. 2019.
- [10] K. C. Okafor, "Dynamic reliability modeling of cyber-physical edge computing network," *Int. J. Comput. Appl.*, vol. 43, no. 7, pp. 612–622, Aug. 2021.
- [11] K. C. Okafor, I. E. Achumba, G. A. Chukwudebe, and G. C. Ononiwu, "Leveraging fog computing for scalable IoT datacenter using spine-leaf network topology," *J. Electr. Comput. Eng.*, vol. 2017, pp. 1–11, Apr. 2017.
- [12] A. A. Mutlag, M. K. A. Ghani, N. Arunkumar, M. A. Mohammed, and O. Mohd, "Enabling technologies for fog computing in healthcare IoT systems," *Future Gener. Comput. Syst.*, vol. 90, pp. 62–78, Jan. 2019.
- [13] O. Faust, Y. Hagiwara, T. J. Hong, O. S. Lih, and U. R. Acharya, "Deep learning for healthcare applications based on physiological signals: A review," *Comput. Methods Programs Biomed.*, vol. 161, pp. 1–13, Jul. 2018.
- [14] X. Zhao, K. Yang, Q. Chen, D. Peng, H. Jiang, X. Xu, and X. Shuang, "Deep learning based mobile data offloading in mobile edge computing systems," *Future Gener. Comput. Syst.*, vol. 99, pp. 346–355, Oct. 2019.
- [15] T.-S. Lim, W.-Y. Loh, and Y.-S. Shih, "A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms," *Mach. Learn.*, vol. 40, no. 3, pp. 203–228, 2000.
- [16] P. Datta and B. Sharma, "A survey on IoT architectures, protocols, security and smart city based applications," in *Proc. 8th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2017, pp. 1–5.
- [17] V. H. Buch, I. Ahmed, and M. Maruthappu, "Artificial intelligence in medicine: Current trends and future possibilities," *Brit. J. Gen. Pract.*, vol. 68, no. 668, pp. 143–144, Mar. 2018.
- [18] T. Panch, H. Mattie, and L. A. Celi, "The 'inconvenient truth' about AI in healthcare," *NPJ Digit. Med.*, vol. 2, no. 1, pp. 1–3, Dec. 2019.
- [19] L. E. Juarez-Orozco, O. Martinez-Manzanera, F. M. Van Der Zant, R. J. J. Knol, and J. Knuuti, "241 deep learning in quantitative PET myocardial perfusion imaging to predict adverse cardiovascular events," *Eur. Heart J., Cardiovascular Imag.*, vol. 20, Jun. 2019, Art. no. jez145.005.
- [20] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, "Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals," *Inf. Sci.*, vol. 415, pp. 190–198, Nov. 2017.
- [21] I. Azimi, J. Takalo-Mattila, A. Anzanpour, A. M. Rahmani, J.-P. Soininen, and P. Liljeberg, "Empowering healthcare IoT systems with hierarchical edge-based deep learning," in *Proc. IEEE/ACM Int. Conf. Connected Health, Appl., Syst. Eng. Technol.*, Sep. 2018, pp. 63–68.
- [22] S. R. Moosavi, T. N. Gia, E. Nigussie, A. M. Rahmani, S. Virtanen, H. Tenhunen, and J. Isoaho, "End-to-end security scheme for mobility enabled healthcare Internet of Things," *Future Gener. Comput. Syst.*, vol. 64, pp. 108–124, Nov. 2016.
- [23] S. Tuli, R. Mahmud, S. Tuli, and R. Buyya, "FogBus: A blockchain-based lightweight framework for edge and fog computing," *J. Syst. Softw.*, vol. 154, pp. 22–36, Aug. 2019.
- [24] C. Vecchiola, X. Chu, and R. Buyya, "Aneka: A software platform for .NET-based cloud computing," *High Speed Large Scale Sci. Comput.*, vol. 18, no. 3, pp. 267–295, 2009.
- [25] S. Ali and M. Ghazal, "Real-time heart attack mobile detection service (RHAMDS): An IoT use case for software defined networks," in *Proc. IEEE 30th Can. Conf. Electr. Comput. Eng. (CCECE)*, Apr. 2017, pp. 1–6.
- [26] M. Rajasekaran, A. Yassine, M. S. Hossain, M. F. Alhamid, and M. Guizani, "Autonomous monitoring in healthcare environment: Reward-based energy charging mechanism for IoMT wireless sensing nodes," *Future Gener. Comput. Syst.*, vol. 98, pp. 565–576, Sep. 2019.
- [27] N. Constant, D. Borthakur, M. Abtahi, H. Dubey, and K. Mankodiya, "Fog-assisted wIoT: A smart fog gateway for end-to-end analytics in wearable Internet of Things," 2017, *arXiv:1701.08680*.
- [28] A. Rajkomar et al., "Scalable and accurate deep learning with electronic health records," *NPJ Digit. Med.*, vol. 1, no. 1, pp. 1–10, 2018.
- [29] T. N. Gia, M. Jiang, V. K. Sarker, A. M. Rahmani, T. Westerlund, P. Liljeberg, and H. Tenhunen, "Low-cost fog-assisted health-care IoT system with energy-efficient sensor nodes," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2017, pp. 1765–1770.
- [30] O. Akrivopoulos, I. Chatzigiannakis, C. Tselios, and A. Antoniou, "On the deployment of healthcare applications over fog computing infrastructure," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, vol. 2, Jul. 2017, pp. 288–293.
- [31] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "GRAM: Graph-based attention model for healthcare representation learning," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 787–795.

- [32] R. K. Barik, R. Priyadarshini, H. Dubey, V. Kumar, and K. Mankodiya, "FogLearn: Leveraging fog-based machine learning for smart system big data analytics," *Int. J. Fog Comput.*, vol. 1, no. 1, pp. 15–34, 2018.
- [33] M. Pham, Y. Mengistu, H. Do, and W. Sheng, "Delivering home healthcare through a cloud-based smart home environment (CoSHE)," *Future Gener. Comput. Syst.*, vol. 81, pp. 129–140, Apr. 2018.
- [34] P. K. Sahoo, S. K. Mohapatra, and S.-L. Wu, "SLA based healthcare big data analysis and computing in cloud network," *J. Parallel Distrib. Comput.*, vol. 119, pp. 121–135, Sep. 2018.
- [35] M. G. R. Alam, M. S. Munir, M. Z. Uddin, M. S. Alam, T. N. Dang, and C. S. Hong, "Edge-of-things computing framework for cost-effective provisioning of healthcare data," *J. Parallel Distrib. Comput.*, vol. 123, pp. 54–60, Jan. 2019.
- [36] R. M. Abdelmoneem, A. Benslimane, E. Shaaban, S. Abdelhamid, and S. Ghoneim, "A cloud-fog based architecture for IoT applications dedicated to healthcare," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [37] O. Akrivopoulos, D. Amaxilatis, A. Antoniou, and I. Chatzigiannakis, "Design and evaluation of a person-centric heart monitoring system over fog computing infrastructure," in *Proc. 1st Int. Workshop Hum.-Centered Sens., Netw., Syst.*, Nov. 2017, pp. 25–30.
- [38] P. Chithaluru, R. Tiwari, and K. Kumar, "Performance analysis of energy efficient opportunistic routing protocols in wireless sensor network," *Int. J. Sensors, Wireless Commun. Control*, vol. 11, no. 1, pp. 24–41, May 2021.
- [39] R. Tiwari, M. Mittal, S. Garg, and S. Kumar, "Energy-aware resource scheduling in FoG environment for IoT-based applications," in *Energy Conservation Solutions for Fog-Edge Computing Paradigms*. Singapore: Springer, 2022, pp. 1–19.
- [40] P. K. Donta, S. N. Srirama, T. Amgoth, and C. S. R. Annavarapu, "Survey on recent advances in IoT application layer protocols and machine learning scope for research directions," *Digit. Commun. Netw.*, Oct. 2021, doi: 10.1016/j.dcan.2021.10.004.
- [41] Y. Li and W. Chen, "A comparative performance assessment of ensemble learning for credit scoring," *Mathematics*, vol. 8, no. 10, p. 1756, Oct. 2020.
- [42] D. K. Sanghera, L. Been, L. Ortega, G. S. Wander, N. K. Mehra, C. E. Aston, J. J. Mulvihill, and S. Ralhan, "Testing the association of novel meta-analysis-derived diabetes risk genes with type II diabetes and related metabolic traits in Asian Indian Sikhs," *J. Hum. Genet.*, vol. 54, no. 3, pp. 162–168, Mar. 2009.
- [43] N. Kato et al., "Trans-ancestry genome-wide association study identifies 12 genetic loci influencing blood pressure and implicates a role for DNA methylation," *Nature Genet.*, vol. 47, no. 11, pp. 1282–1293, 2015.
- [44] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Int. Workshop Multiple Classifier Syst.*, 2000, pp. 1–15.
- [45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.
- [46] D. Dua and C. Graff, "UCI machine learning repository," Univ. California, School Inf. Comput. Sci., Irvine, CA, USA, Tech. Rep. (714) 856-8779, 2019. Accessed: Sep. 14, 2019. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Heart+Disease>
- [47] R. Malik et al., "Multiancestry genome-wide association study of 520,000 subjects identifies 32 loci associated with stroke and stroke subtypes," *Nature Genet.*, vol. 50, no. 4, pp. 524–537, 2018.



PARAG VERMA having experience of more than nine years in the field of industry and academics. He is having more than 25 international papers in reputed conferences and journals. He has contributed three books with publishers like Taylor and France (CRC Press), Nerosa and Alpha, and currently working on two more accepted books. He had also contributed more than five chapters for reputed publishers. He is associated with many societies and organizations for the welfare of educationalist societies. His research interests include the IoT, healthcare, and image processing-based analysis using distributed computing platforms. He is also an Editorial Board Member of many reputed conferences and journals, including Scopus. He is also a Guest Editor of *IJNCDS* (Scopus Indexed Inderscience Journal) and many more.



RAJEEV TIWARI (Senior Member, IEEE) received the Ph.D. degree from Thapar University, Patiala. He has more than 15 years of research and teaching experience. He has published different articles in international journals of repute with Elsevier, Springer, and IEEE. He has chaired many international conferences globally and delivered keynotes in field of sensor network and the IoT-fog-cloud-based computing. He has edited four books with Springer and Taylor and Francis. He has completed one project on smart parking system using the IoT devices. He has three patents by his name. He is a Senior Member of IEIM.



WEI-CHIANG HONG (Senior Member, IEEE) is currently a Professor with the Department of Information Management, Asia Eastern University of Science and Technology, Taiwan. His articles have been published in *Applied Mathematics and Computation*, *Applied Soft Computing*, *Decision Support Systems*, *Energy*, *Energy Policy*, *European Journal of Operational Research*, the IEEE TRANSACTIONS ON FUZZY SYSTEMS, and *Neurocomputing* among others. His research interests include computational intelligence and application of forecasting technology. He is a Senior Member of IIE. He is also on the editorial board of several journals, including *International Journal of Applied Evolutionary Computation*, *Neurocomputing*, and *Applied Soft Computing*. In May 2012, his paper had been evaluated as "Top Cited Article 2007–2011" by Elsevier Publisher (The Netherlands). In October 2021, he was indexed in World's Top 2% Scientists (issued by Stanford University).



SHUCHI UPADHYAY is currently pursuing the Ph.D. degree in food and nutrition. She is also working as an Assistant Professor with the Department of Allied and Health Sciences, School of Health Science and Technology, UPES, Dehradun. She has experience of clinical trials analysis and experimentation. She has more than 14 years of research and teaching experience. She has many papers in international renowned journals, conferences, and book chapters. She has completed one Government of India Funded Project with the Baal Vikas Department. She is also working in another funded project as a CO PI in healthcare and pharma area.



YI-HSUAN YEH is currently an Assistant Professor with the Department of Information Management, Asia Eastern University of Science and Technology, Taiwan. Her articles have been published in *Utilities Policy*, *Applied Soft Computing*, and among others. She has more than 20 years of research and teaching experience. She has completed more than ten enterprises' projects in Taiwan. Her research interests include computational intelligence and application of information technology.