

Healthcare Monitoring Using Ensemble Classifiers in Fog Computing Framework

P. M. Arunkumar¹, Mehedi Masud², Sultan Aljahdali² and Mohamed Abouhawwash^{3,4,*}

¹Department of Computer Science and Engineering, Karpagam College of Engineering, Coimbatore, 641032, India

²Department of Computer Science, College of Computers and Information Technology, Taif University, P. O. Box 11099, Taif, 21944, Saudi Arabia

³Department of Mathematics, Faculty of Science, Mansoura University, Mansoura, 35516, Egypt

⁴Department of Computational Mathematics, Science, and Engineering (CMSE), Michigan State University, East Lansing, MI, 48824, USA

*Corresponding Author: Mohamed Abouhawwash. Email: abouhaww@msu.edu

Received: 22 May 2022; Accepted: 27 June 2022

Abstract: Nowadays, the cloud environment faces numerous issues like synchronizing information before the switch over the data migration. The requirement for a centralized internet of things (IoT)-based system has been restricted to some extent. Due to low scalability on security considerations, the cloud seems uninteresting. Since healthcare networks demand computer operations on large amounts of data, the sensitivity of device latency evolved among health networks is a challenging issue. In comparison to cloud domains, the new paradigms of fog computing give fresh alternatives by bringing resources closer to users by providing low latency and energy-efficient data processing solutions. Previous fog computing frameworks have various flaws, such as overvaluing response time or ignoring the accuracy of the result yet handling both at the same time compromises the network community. In this proposed work, Health Fog is integrated with the Optimized Cascaded Convolution Neural Network framework for diagnosing heart disease. Initially, the data is collected, and then pre-processing is done by Linear Discriminant Analysis. Then the features are extracted and optimized using Galactic Swarm Optimization. The optimized features are given into the Health Fog framework for diagnosing heart disease patients. It uses ensemble-based deep learning in edge computing devices, which automatically monitors real-life health networks such as heart disease analysis. Finally, the classifiers such as bagging, boosting, XGBoost, Multi-Layer Perceptron (MLP), and Partitions (PART) are used for classifying the data. Then the majority voting classifier predicts the result. This work uses FogBus architecture and evaluates the execution of power usage, bandwidth of the network, latency, execution time, and accuracy.

Keywords: Healthfog; FogBus; cascaded convolution neural network; cloud computing; heart disease; automatic health monitoring; internet of things



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

Fog and Cloud general-purpose computing has developed as the foundation of the market world, relying on the Internet to supply customers with on-demand products. Each of these fields has attracted a lot of interest from industry and academics. Cloud computing, on the other hand, is not a good alternative for programs that demand real-time reaction because of the substantial time latency. Edge computing, fog computing, the Internet of Things (IoT), and Big Data are examples of how technological advances have gained traction. Due to their durability and capacity, the variety of performance parameters depends on the intended applications [1].

Cloud and fog computing concepts have gotten a lot of attention and have become a foundation again for the market world, which relies on Internet services to supply customers with on-demand operational resources. Both industry and academia have adopted these subjects as vital components. Due to the considerable time delay, cloud computing is not ideal for implementations to get answers [2]. Modern innovations such as Big Data, fog computing, the Internet of Things, and edge computing have exploded in popularity due to their capability to provide a variety of performance parameters focused on the target workloads.

In the world of professionals and academia, those two variables have gotten a lot of attention. However, because of the significant delay in reaction time, cloud technology is not a good choice for programs that demand constant feedback. Because of their characteristics as warm-heartedness and capacity to offer reaction attributes. Depending on the monitored desired applications, developments and improvements such as big data management with the Internet of Things (IoT), Fog computing, and Edge computing have become crucial [3].

Fog computing flawlessly coordinates delay and consistent implementations thanks to high volume data storage, calculation, and efficient communication practices. That is supplied by technological innovations such as edge devices that motivate and enhance movement, safety issues, safety, smaller-scale delay, and network bandwidth. [4–7].

Cloud computing is actively supporting the development of program guidelines, for instance, via IoT systems, fog nodes, cloud technologies, and big data management and architecture [8,9]. Fog computing employs routers, switches, compute nodes, and gateways to provide the least amount of help with power usage, networking inactivity, and network speed of response. Deep learning has also been used to forecast and categorize health information with extraordinarily high precision [10]. Current deep learning methods for healthcare systems, on the other hand, are quite complex and need a significant amount of computing power capacity for both training and testing [11]. This also takes a while to learn and interpret information with these complex brain networks. Evermore advanced the system and the longer the prediction period, the higher the quality required. It has been a significant issue in health and other IoT applications wherein real-time findings are crucial.

The main contribution of the proposed work is given below:

- Integrating HealthFog and Optimized Cascaded Convolution Neural Network provides an efficient smart diagnosis of heart disease.
- The proposed method is deployed using the FogBus structure which includes IoT-Edge-Computing devices for real-time examination.
- This optimization aids in increasing accuracy while lowering error rates.
- It uses Ensemble-based deep learning models for solving a binary classification issue.

The rest of our research article is written as follows: Section 2 discusses the related work on HealthFog, ensemble models, fog computing, and deep learning models. Section 3 shows the general working

methodology of the proposed work. Section 4 evaluates the implementation and results of the proposed method. Section 5 concludes the work and discusses the result evaluation.

2 Related Work

The fog computing structure is a new technology for efficiently accessing medical information from a multitude of IoT devices. While edge devices are nearer to IoT devices than cloud computing environments, fog computing can process data from cardiac patients at edge devices or fog nodes using enormous processing capacity, access speed, reaction time, and latency. Fog computing forms a key concept for the proper organization of healthcare data in the medical field. It can be acquired via various IoT-enabled devices. Edge computing-enabled networks are significantly closer to IoT-enabled devices than cloud-based data centers. So, fog computing compatible devices or fog nodes for measuring gauze can deal with heart patients' data, drastically lowering delay, latency, or speed of response.

For collecting the medical information from various cardiac patients, the author [12] presented a Low-Cost Health Monitoring (LCHM) paradigm. Furthermore, sensor nodes track and analyze the Electro Cardio Graphy (ECG) in real-time to organize and manage heart patient data, whereas LCHM has a longer reaction time, lowering efficiency. Furthermore, sensor nodes collect ECG, respiratory rate, and temperature and communicate them to an intelligent gateway via wireless transmission mode, allowing the intelligent gateway to make an autonomous choice to assist the patient rapidly. The effectiveness of the LCHM framework in order of processing time is tested using an Orange Pi One-based small-scale testbed, but LCHM produces more power throughout data collecting and communication.

In [13], the author introduced FogCepCare, an IoT-based health information management paradigm that integrates the cloud layer with the sensor layer to determine the overall health of cardiac patients and decreases task processing completion time in real-time. To reduce completion time, FogCepCare employs a segmentation and grouping strategy, as well as a messaging and parallel computing policy. The efficiency of FogCepCare is evaluated to that of existing models in a modeled cloud infrastructure, and it optimizes completion time. However, this research lacked an evaluation of the results in terms of crucial QoS factors like power usage, delay, and precision.

The author [14–18] presented an IoT e-health strategy focused on a Software Defined Network (SDN) program that gathers information via smartphone voice activation and determines patients' overall health. Furthermore, an IoT e-health service uses a smartphone application concept to determine the type of heart attack; however, the suggested user's functionality is not tested in public clouds.

To study the possibility of implementing the Convolutional Neural Network (CNN) dependent classifier model as an instance of deep learning approaches, the researcher presented a Hierarchical Edge-based deep learning (HEDL) based healthcare IoT platform.

The author [19–21] suggested a Fog-based IoT-Healthcare (FIH) effective construction and investigated the incorporation of Cloud computing in interoperability Clinical services that went beyond the typical Cloud-based model. The various models discussed above have major issues in the data flow, latency, etc. Innovative fog-based architecture is required to process the data with security and latency.

3 Framework Automation

FogBus [22] is a tool for building and deploying combined Fog-Cloud scenarios with organized communication and application execution that is easily configurable. FogBus links a variety of IoT sensors, such as healthcare sensors, to gateways equipment to communicate data and duties to fog worker nodes. Fog broker nodes handle resource utilization and task execution. FogBus combines blockchain, identification, and encrypted mechanisms to assure the integrity of data, confidentiality, and safety,

increasing the fog atmosphere's accuracy and performance. FogBus communicates over (Hypertext Transfer Protocol) HTTP RESTful Application Programming Interface (APIs) and easily combines fog installation with the Cloud via the Aneka software system.

Aneka [23] is a cloud-based technology platform and guidelines for building and deploying application programs. Designers can use Aneka's APIs to access virtual cloud services. The Aneka platform's fundamental components are service-oriented in formulation and construction. The capacity to dynamically purchase assets and incorporate these into traditional networks and software platforms is known as dynamic supply. Virtual Machines (VMs) purchased from an Infrastructure-as-a-Service (IaaS) cloud platform is the most frequent assets. Provision solutions for distributing virtual units from cloud services to supplement local resources are granted based on the Fabric Services in Aneka.

Models include the Bag of Jobs, Shared Threads, MapReduce, and Parameters Sweeps. For job distribution among cloud VMs in HealthFog and FETCH, we were using the Bag of Tasks paradigm. FogBus is used to capture fog services, and Aneka is used to capture the public cloud in HealthFog-FE [24–40].

4 System Architecture

The Health Fog-CCNN framework is based on IoT, fog-enabled cloud-based software for health data. It successfully manages the data from heart patients and analyzes their medical conditions to assess the extent of heart disease. Utilizing software applications, Health Fog-CCNN combines a variety of physical devices, allowing for a systematic and flawless end-to-end connection of Edge-Fog-Cloud for quick and precise findings transmission. Fig. 1 depicts the Health Fog CCNN design, which is made up of several software and hardware components that will be discussed later.

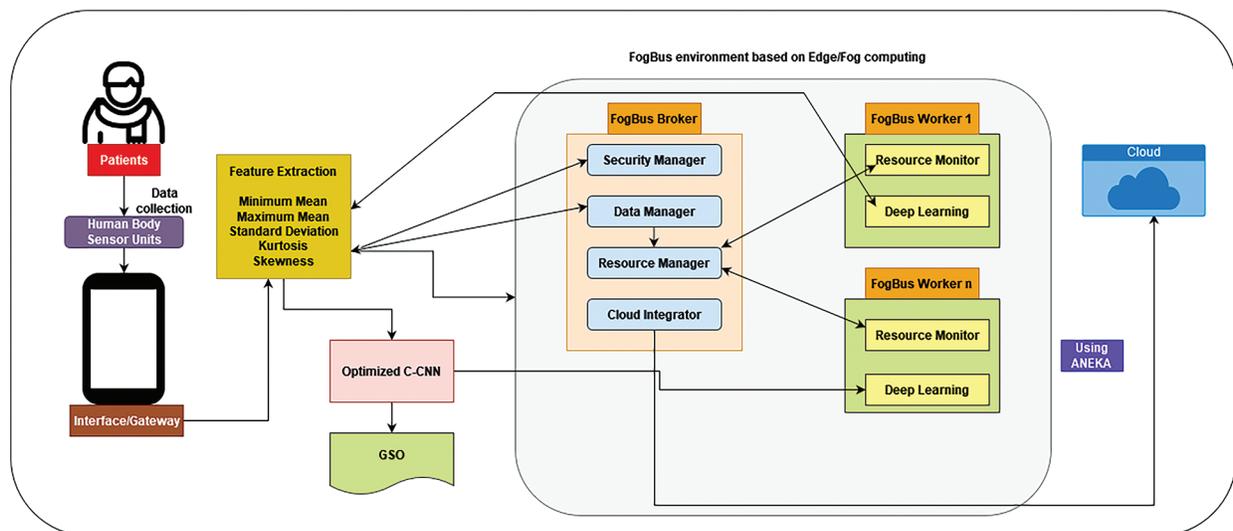


Figure 1: Architecture of health fog-CCNN

Initially, the data is collected, and then pre-processing is done by Linear Discriminant Analysis. Then the features are extracted and optimized using Galactic Swarm Optimization (GSO). The optimized features are given into the Health Fog framework for diagnosing heart disease patients. It uses ensemble-based deep learning in edge computing devices, which automatically monitors real-life health networks such as heart

disease analysis. Finally, the classifiers such as bagging, boosting, XBoost, MLP, and PART are used for classifying the data. Then the majority voting classifier predicts the result.

4.1 Hardware Equipment Used for HealthFog-CCNN

The proposed HealthFog-CCNN contains the following hardware equipment. They are Body Area Sensor, Gateway used for Application, and FogBus framework.

4.1.1 Networks Based on Body Area Sensor (BAS)

This category is defined by three kinds of sensors: medical sensors, activities sensors, and environmental sensors. This section can identify the information of heart patients and send it to the appropriate route devices.

4.1.2 Gateway Used for Applications

The suggested system uses three distinct kinds of gadgets for the application gateway, including smartphones, computers, and pads, which act as fog-enabled tools to gather data from various sensors and pass it to the Brokerage node for further analysis.

4.1.3 FogBus Framework

A FogBus framework consists of three parts, nodes used for the worker, broker, and cloud-based data center.

The node used for Broker

This section gets the work requirement or possible input data from gateway devices. The labor demands are received by the solicitations input section from gateway devices which does not stay long until data is sent. The assertions modules (a component of the task scheduler in the delegation node) collect all the stakes information management from the specialized networks and decide which nodes or sub-node to transmit to the enterprises in stages.

The node used for Worker

This section focuses on completing the initiatives authorized by the resource development of the broker's nodes. Gadgets and single-board processors (SBCs) like the Raspberry Pi processor might be mounted on the worker's nodes. Worker nodes in the HealthFog CCNN can use new in-depth learning approaches to evaluate data and develop and measure outcomes. Different elements for processing data, information extraction and mining, Big Data analytics, and storage can be added to the Worker node. The Worker nodes receive input information directly from the Gateway equipment, produce outputs, and exchange them with the Gateway gadgets. The Broker node in the health-Fog model can also act as a Worker component.

Cloud-Based Data Center

Whenever the fog computing framework gets congested, the availability of services suffers as a result of the delay, or the bulk of the data is substantially bigger than typical, HealthFog-CCNN enterprises turn to the cloud-based data center for help.

4.2 Software Elements of the HealthFog-CCNN Method

The HealthFog-CCNN method consists of the following process. After collecting the data from the patients, the data is pre-processed. Then the Linear Discriminant Analysis is used for filtering.

4.2.1 Pre-Processing and Data Filtering

Pre-processing the information is the first stage once it is entered. This involves employing data analytics tools to filter information. The main goal of retrieving key elements of information extracted features that

impact the health condition of the patient, the filtered data is limited to a shorter length utilizing Linear Discriminant Analysis employing Set Partitioning in Hierarchical Trees (SPIHT) algorithm and encrypted using Singular Value Decomposition (SVD) method. It effectively makes a judgment extracted from different information, which prescribes medications and appropriate check-ups based on the process of training examples of health professionals and physicians and keeps it in a database for re-training as necessary.

Linear Discriminant Analysis (LDA)

Linear discriminant analysis is a reliable classification technique that may also be used for data presentation and dimension reduction. This is a supervised machine learning technique that calculates boundary that improves the distinction among several categories employed, unlike Principal Component Analysis (PCA), which aims to maximize variation.

By maximizing intervals among predicted averages and limiting predicted variation, it attempts to divide various classes. These optimization techniques are integrated into single criterion functions that can be stated as follows for binary classification:

$$\max_{\omega} J(\omega) = \frac{(M_1 - M_2)^2}{s_1^2 + s_2^2} \quad (1)$$

Here $(M_1 - M_2)^2$ is the means distinction among the 2 groups and $s_1^2 + s_2^2$ is the overall scattering (standard deviation (sd) of the 2 classes. LDA aims to discover the vectors w that maximizes the criterion function $J(w)$.

In LDA, all K categories are considered to have the same covariance. We may generate the following discriminant function for the k th class using this assertion:

$$\delta_k(x) = x^T \sum \mu_k - \frac{1}{2} \mu_k^T \sum -1 \mu_k + \log \pi_k \quad (2)$$

Classes are separated as much as feasible from one another, and characteristics inside a class are kept as close together as practicable. The dissociation capability of converted dimensions is used to rank them. The maximum number of items must be one less than that of the number of categorization categories. As a result, because this was binary classifier research, we simply used the very first linear discriminant.

4.2.2 Feature Extraction

For feature extraction it uses min and max means, standard deviation, kurtosis, and skewness. It helps to solve the over fitting issues and number of repetitive data.

Min and Max mean

“Attaching the input data and dividing it by the total of data values” yields the mean for a dataset.

$$\mu = \frac{\sum Y_d}{d} \quad (3)$$

All of the sets of numbers are written as Y_d , while the total number of values is written as d . The lowest value in the information is known to as the mean of the upper and lower bounds, and the biggest value in the dataset is known to as the mean of the upper and lower bounds, correspondingly.

Standard Deviation

This is “the sum of the squares of variability calculated by calculating every information point’s dispersion from the mean.” The standard deviation formula is found in Eq. (4).

$$\sigma^2 = \frac{1}{N} \sum_{j=0}^{N-1} (y_j - \mu)^2 \quad (4)$$

Kurtosis

As defined in “measurements of the data are heavy-tailed or light-tailed respect to normally distributed, “it is” statistical measures of how close are heavy-tailed or light-tailed compared to a normally distributed.”

$$Ku = \frac{1}{D} \sum_{d=1}^D \left[\frac{Y_d - \bar{Y}}{\sigma} \right]^4 \quad (5)$$

Skewness

This “corresponds to a distortion or asymmetrical in a data set that differs from symmetrical bell-shaped curve, or normal distribution,” as defined in

$$Ku = \frac{1}{D} \sum_{d=1}^D \left[\frac{Y_d - \bar{Y}}{\sigma} \right]^3 \quad (6)$$

4.2.3 Resource Management

Workload management and arbitration element are the two components that make up this system. For processing data, the workload management keeps track of job requests and task queues. It also manages huge amounts of data that must be analyzed. The Arbitration component allocates the fog or cloud services that have been supplied for the execution of tasks that have been prioritized and managed by workload management. The Arbitration modules is located in the Broker node and determines whether Fog computing node, the Broker, the Fog worker node, or the Cloud Data Center, must be given the information to acquire the outcomes.

The main goal is to split the tasks into various resources in order to handle the stacks and provide optimal performance. Users of HealthFog CCNN have the ability to customize arbitral projections based on their own loading balance and program needs. Fig. 2 depicts the suggested scheme as a flow chart, and the pseudo code is shown in Algorithm 1.

Algorithm 1: HealthFog-based resource scheduling pseudo-code

Input: TA_n , WO_{L1} , WO_{L2}, \dots, WO_{Ln}

Output: stores the tasks in the cloud

1. Evaluate $TA_w = \min (WO_{L1}, WO_{L2}, \dots, WO_{Ln})$
 2. if $TA_w > T$,
 3. else
 4. handover TA_n to argmin TA_w
 5. end
-

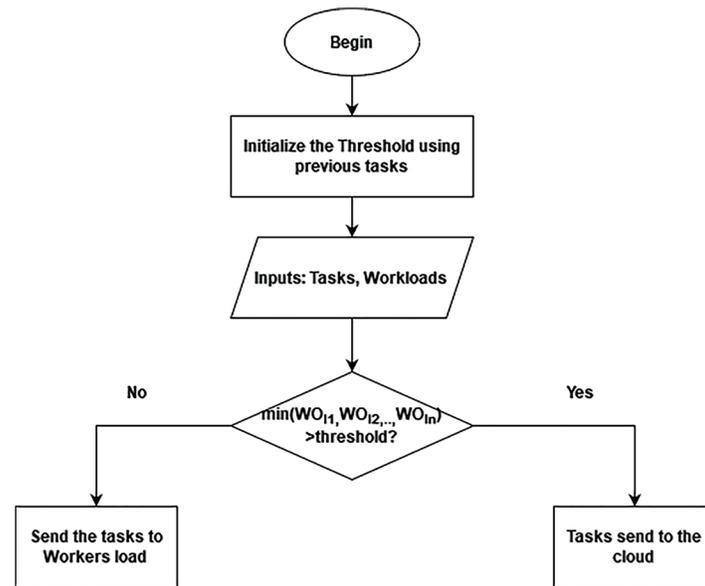


Figure 2: Flowchart of health got based on resource scheduling

4.2.4 Ensemble Module

This system utilizes a majority voting approach to determine the yielding category which would be needed if the patient has a cardiac disease, and it is combined with a deep learning model to boost the predicted results from multiple models. The majority vote basis accumulates the learners' outcomes in the same way that the weighted average does. The majority voting base, on the other hand, collects the learners' votes and forecasts the ending labeling as the label with more votes, rather than calculating the mean of the probabilities outcomes.

Since the quantity of majority votes surpasses the effects, the majority vote is less skewed toward the defined point learner's outcomes than the weighted average. In the ensemble model, meanwhile, event domination is caused by the majority of the same weak learners or relying on base learning to select a specific event. The sum of weak learners (W_1) is obtained as of Eq. (3)

$$Su_L(\cdot) = \arg \max_k [card(l|W_l(\cdot) = k)] \quad (7)$$

This component is capable of propagating data and providing findings from many other nodes of workers in the FogBus hub, which would be delivered to the assignments.

4.3 Structure of HealthFog-CCNN

The HealthFog-CCNN elements previously explained transmit a considerable quantity of information, knowledge, and control signals. This is expected to enable this steady network connectivity. Furthermore, the transmission must be consistent and fault-tolerant. With all of this in mind, the elements are organized in the topology depicted in Fig. 1. FogBus is used to promote communication between all machines on the Edge, while Aneka is used to improve communication with Cloud VM.

The HealthFog-CCNN device's system design is based on a master-slave structure, with each broker acting as a worker to assess and manage the workload. Within cases of nonscenario, the gateway worker/broker delivers work to the example input data for the probes to execute, during which time the preprocessing, prediction model, and computed results are sent back to the gateway workers. Because the gateway gadgets might connect to the virtual private network (VPN) in the terms of cloud sending,

the data is sent to a node of brokers, who then transfer the data collected to the CDC. It also ensures that hostile components and programmers can't access the IoT sensors and gateway devices because they can't communicate to the web-based system; instead, just local area networks (LANs) can communicate to other nodes. The duration of the operation is reliant upon both broker's nodes as well as the CDC obtaining increased connection expense and much less lazy owing to the latencies delay because the cloud-based platform can access a vast range of resources. When you get to this point, the organization is more powerful, and the data it receives is more accurate. Any residual edge is forwarded to the broker/worker nodes and the specialized node picks up a big amount of data transmitted via the bagging procedure. All edge computing is included in the HealthFog-CCNN system. Enabled devices, such as gateway devices and network nodes operating as a broker and sharing a LAN as the resource for learning. The software section broker node houses the manager. As a result, devices for broker job demands have emerged as a doorway. In the form of workers clouds that receives work requests. The resource manager's discretion result is as follows: collected through the use of an access point that delivers data on where the info would have been sent. There are three possibilities here:

- Data to a worker node is handled by a broker.
- Additional worker node for data transfer.
- The CDC (Cloud-based Data Center) is used to process information.

The default gateway might deliver data right to the worker's node or indeed the broker, based on the circumstances. Unless the agency seems to have enough connections and the workers nodes are overloaded can the agent enable the specified administrator to transmit information. If data must be transferred to the cloud, it has to go through the broker's node because gateways are unable to connect to a VPN with a cloud-based virtual environment.

4.4 Deep Learning Module

4.4.1 Optimized Cascaded CNN

For predicting heart disorders, the suggested model employs the CCNN [2]. It takes into account into data properties. The GSO algorithm optimizes the layers of the cascaded network, hidden neurons, and activation function of the CCNN as a novel improvement to the original CCNN. It leads to a better diagnosis of heart disorders and a greater predictive accuracy with lower error margins.

The entropy losses are used to represent the cascaded CNN architecture and the layers in the cascaded networks are given a threshold value. The input, which includes data characteristics, is first sent to CNN's convolution layer, which is then transmitted to the max pooling.

Hidden neurons (HNe) are really the CCNN's characteristics, and every layer's activation function (AF) can be set. The hidden neuron range is allocated between 5 and 255. The limitation of AF is provided As compared to other means, Rectified Linear Unit (ReLU) has more advantages since it does not stimulate complete neurons at the same moment.

The objective of this project's intelligent health system using the GSO method is to reduce the Mean Square Error (MSE) between projected and real results, as shown in Fig. 1.

$$obj = argmin(MSE), \{af, h_{NE}\} \quad (8)$$

It can be used for "finding the mean of the squares of the mistakes, which is the mean squared variance between the projected results Pr_q and the actual results Ac_q ," according to the specification.

$$MSE = \frac{1}{q} \sum_{q=i}^q (Ac_q - Pr_q)^2 \quad (9)$$

The entire quantity of characteristics is referred to as q . Thus, the reduction of error leads to a higher predictions rate for the automated health system with IoT-assisted Healthfog-edge computing.

4.4.2 GSO

The GSO [2] method is used in the suggested automated healthcare model with IoT-assisted Healthfog-edge computing to improve the prediction rate of cardiac disorders. It is employed in the optimization of the cascaded network's hidden layers, hidden neurons, and transfer functions. This improvement aids in increasing accuracy while lowering error rates. The GSO algorithm was chosen because of its several advantages, including the ability to obtain local optimal solutions, a quick convergence rate, the ability to identify local solutions in order to obtain the optimum solution, and correct balances between both the exploiting and exploratory stages.

Moreover, the problem of converging to a local optimal solution is addressed during in the discovery phase, increasing the speed convergence speed than other previous techniques. Fig. 3 shows the optimized CCNN process for Heart disease prediction.

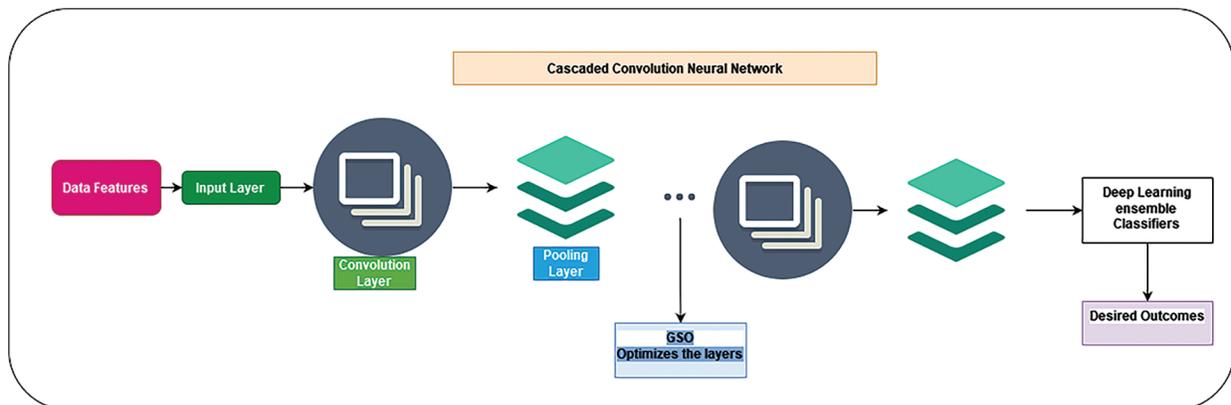


Figure 3: Optimized CCNN process

4.4.3 Classifiers Used in Deep Learning Based Ensemble Methods

In this HealthFog-FE method, deep learning-based ensemble classifiers were used for binary classification issues. It uses Bagging, Multilayer Perceptron, Boosting, Partial Decision tree and Gradient Boosted Decision tree (XGboost) for classifying the data. The method is initially trained using the Cleveland Dataset's cardiac patient information and the associated known output class, and then utilized to forecast actual data inputs outputs, as illustrated in Fig. 4.

5 Result Analysis

Python was used to develop the pre-processing and ensemble deep learning elements. The pre-processing component normalizes the information using the dataset's min and max fields parameters, as well as their distributions. SciKit learns Library was used in the ensemble deep training module [1]. To construct our voting strategy, we used the SciKit learn Library's Bagging, Boosting, MLP, Xboost, and PART Classifier. The approach recognizes as input the type of base classifier, which would in our instance, is a deep neural network, and the number of classifiers. To train the classifiers, the program now

arbitrarily divides the information between them. It receives all forecasted categories as input and produces the majority forecast at diagnostic time. The variables of the best base model on our data set after tuning are listed below. Parameters are listed in [Tab. 1](#).

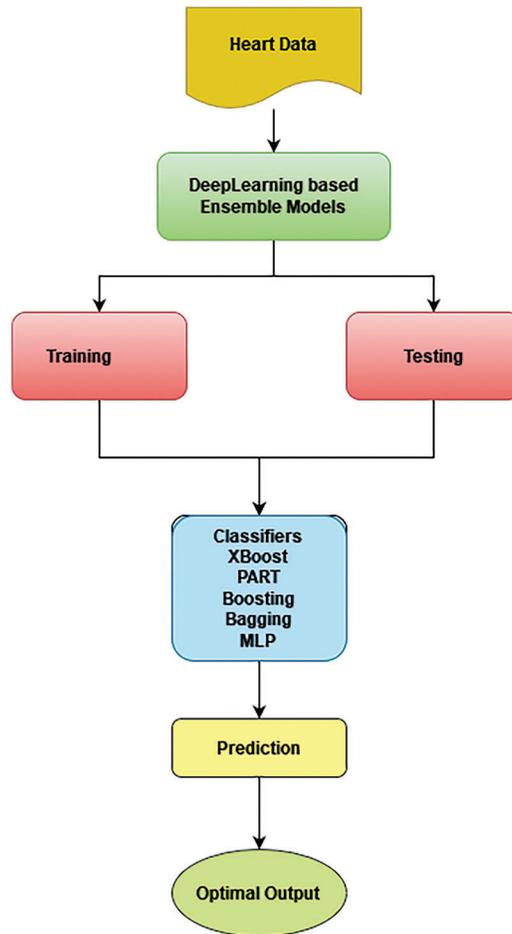


Figure 4: HealthFog-FE training and testing model

Table 1: Parameters used and characteristics

Input layer's size	13
Output layer's size	2
No of hidden layers	3
Layers descriptions	Fully connected (FC) layer with 20 nodes, FC layer with 20 nodes and 10 nodes.
Optimizer used	Adam
Activation function	ReLU

We developed and installed the proposed HealthFog-CCNN model on real Fog architecture of gadgets by using FogBus platform [1] to show its practicality and effectiveness. The method was established in a real-life application to identify heart issues in patients quickly using cutting-edge deep learning methods and a fog computing environment in Tab. 2. We looked at reliability and response times, as well as connection and power expenses, to demonstrate that the HealthFog-CCNN model is efficient.

5.1 Simulation Environment

Table 2: The hardware configurations are given

Gateway device	Samsung galaxy S7 with android 9
Broker/Master Node	Dell XPS 13 with Intel(R) Core(TM) i5-7200 central processing unit (CPU) @ 2.50 GHZ, 8.00 GB DDR4 Random Access Memory(RAM) and 64-bit Windows 10. The deployment used Apache HTTP server 2.4.34.
Worker node	Raspberry Pi 3B+, RISC Machine (ARM) Cortex-A53 quad-core CPU @ 1.4 GHz and 1 GB LPDDR2 SDRAM and IEEE 802.11 Wifi. Raspbian stretch operating system with apache HTTP server 2.4.34.
Public cloud	Microsoft azure B1 s machine, 1vCPU, 1 GB RAM, 2 GB solid-state drive (SSD), Windows Server 2016.

5.2 Dataset Description

We used data from cardiac patients to predict the existence of heart problems in the person, that is an integer number of 0 (no presence) or 1 (presence) (presence). The trials are conducted using the Cleveland database [23], which was produced by Andras Janosi (M.D.) of the Gottsegen Hungarian Institute of Cardiology in Hungary and colleagues. The identities of the patients, as well as their patient numbers, are kept private.

It uses 7:2:1 ratio for evaluation. In the dataset 70% are used for training, 20% are used for testing and 10% are used for validation. The evaluation metrics such as accuracy, execution time, energy consumption, jitter, latency and network bandwidth. The classifiers are used for evaluation. In Tab. 3 the comparison of classifiers is given. The majority voting selects the majority values upon the classifiers. Comparing with all the algorithms MLP outperforms existing models.

Table 3: Evaluation of different classifiers used in healthfog-CCNN

Classifiers	Accuracy (%)	Energy consumption (J)	Latency (millisec)	Bandwidth (Mbps)
Bagging	4.20	920	756.78	880
PART	3.65	635	475.30	858
MLP	7.85	520	978.25	950
Xboost	4.12	810	815.20	901
Boosting	5.78	845	420.65	876

In Fig. 5 shows the accuracy of different classifiers used. Among all the classifiers, MLP outperforms higher accuracy level in the diagnosis of heart disease. By combining HealthFog with CCNN, it improves the accuracy level.

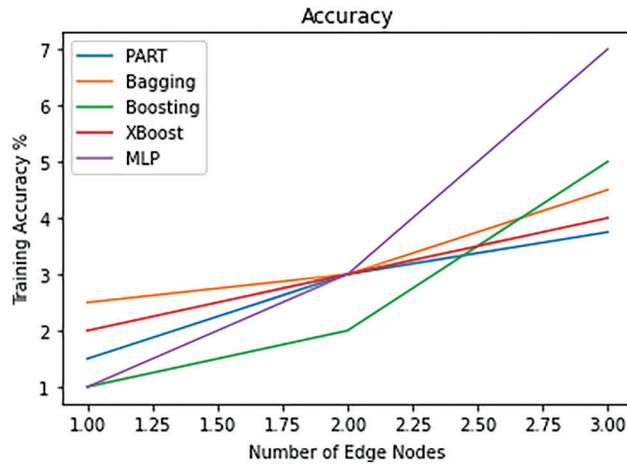


Figure 5: Accuracy

For HealthFog-CCNN, the MLP classifiers achieve better result. Other environments like cloud setup, it takes very high energy consumption. In Fig. 6 it shows the latency of proposed method. The MLP classifier outperforms better compared with all the classifiers.

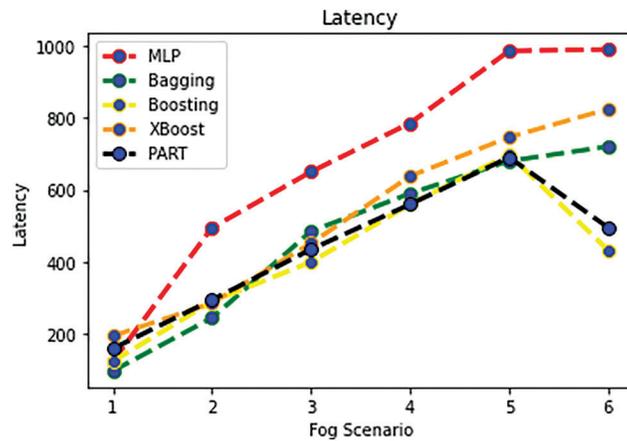


Figure 6: Latency

In Fig. 7 it shows the execution time of proposed work. Compared with all the classifiers the MLP outperforms better. The execution time for HealthFog-CCNN is minimized. Whereas cloud environment takes more time for execution.

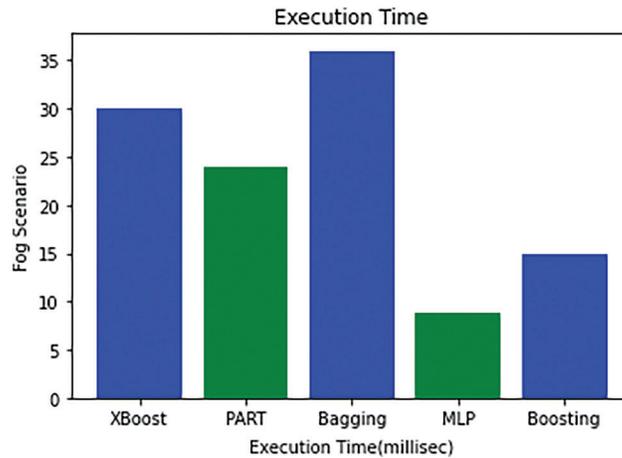


Figure 7: Execution time

6 Conclusion

The health care system as a service is a massive undertaking. In this work, HealthFog is integrated with the Optimized Cascaded Convolution Neural Network framework for diagnosing heart disease. We propose a new Fog-based Smart Healthcare System for Automatic Diagnosis of Heart Diseases utilizing deep learning and IoT dubbed HealthFog-CCNN. This study effort focuses solely on the medical elements for patients with heart disease. HealthFog-CCNN gives service as a fog service and effectively organizes information from numerous IoT devices for heart patients. HealthFog-CCNN incorporates deep learning into Edge computing devices and uses it to analyze Heart Problems in real-time. Previous research on these Heart Patient analyses did not need to use deep learning and had poor predictive performance, rendering them ineffective in real-world situations. Deep learning-based methods that demand a lot of computational resources (CPU and GPU) for both classification and prediction require a lot of computing resources (CPU and GPU). Employing unique communications and modeling distribution strategies like ensembling, this study enabled complicated deep learning networks to be incorporated in Edge computing paradigms, allowing for good accuracy with really low latencies. Training the neural network on a common dataset and creating a functioning system that gives a real-time predictive performance, was also proven for real-life heart patient information processing. In the proposed method we use five different classifiers for diagnosing heart patients. Compared with all the classifiers, MLP outperforms better in all the parameter metrics such as energy usage, communication bandwidth, latency, accuracy, and processing time. In this, HealthFog-CCNN efficiently predicts the heart disease. In future the deep learning can be used for improving the performance of the metrics.

Funding Statement: This work was supported by Taif University Researchers Supporting Project (TURSP) under number (TURSP-2020/73), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] S. Tuli, N. Basumatary, S. S. Gill, M. Kahani, R. C. Arya *et al.*, "HealthFog: An ensemble deep learning-based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments," *Future Generation Computer Systems*, vol. 104, no. 12, pp. 187–200, 2020.

- [2] K. B. Raju, S. Dara, A. Vidyarthi, V. M. Gupta and B. Khan, "Smart heart disease prediction system with IoT and fog computing sectors enabled by cascaded deep learning model," *Computational Intelligence and Neuroscience*, vol. 13, no. 2, pp. 24–45, 2022.
- [3] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi *et al.*, "Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach," *Future Generation Computer Systems*, vol. 78, no. 10, pp. 641–658, 2018.
- [4] S. Tuli, N. Basumatary and R. Buyya, "Edgelens: Deep learning-based object detection in integrated IoT, fog and cloud computing environments," in *Proc. 2019 4th Int. Conf. on Information Systems and Computer Networks (ISCON)*, Mathura, India, pp. 496–502, 2019.
- [5] 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. on Intelligent Data Communication Technologies and Internet of Things*, India, pp. 1376–1383, 2018.
- [6] 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 Communications*, vol. 14, no. 11, pp. 1–16, 2017.
- [7] A. Goyal, K. Narang, G. Ahluwalia, P. M. Sohal, B. Singh *et al.*, "Seasonal variation in 24 h blood pressure profile in a healthy adults: A prospective observational study," *Journal of Human Hypertension*, vol. 33, no. 8, pp. 626–633, 2019.
- [8] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.
- [9] X. Zhao, K. Yang, Q. Chen, D. Peng, H. Jiang *et al.*, "Deep learning-based mobile data offloading in mobile edge computing systems," *Future Generation Computer Systems*, vol. 99, no. 11, pp. 346–355, 2019.
- [10] 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. on Intelligent Data Communication Technologies and Internet of Things*, Mathura, India, pp. 1376–1383, 2018.
- [11] T. N. Gia, M. Jiang, V. K. Sarker, A. M. Rahmani, T. Westerlund *et al.*, "Low cost fog assisted health-care IoT system with energy efficient sensor nodes," in *Proc. 13th Int. Wireless Communications and Mobile Computing Conf. (IWCMC)*, Chennai, India, pp. 1765–1770, 2017.
- [12] 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 Canadian Conf. on Electrical and Computer Engineering (CCECE)*, Ontario, Canada, pp. 1–6, 2017.
- [13] 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. First Int. Workshop on Human-Centered Sensing, Networking, and Systems*, New York, USA, pp. 25–30, 2017.
- [14] D. K. Sanghera, C. Bejar, B. Sapkota, G. S. Wander and S. Ralhan, "Frequencies of poor metabolizer alleles of 12 pharmacogenomic actionable genes in punjabi sikhs of Indian origin," *Scientific Reports*, vol. 8, no. 1, pp. 1–9, 2018.
- [15] M. Rajasekaran, A. Yassine, M. S. Hossain, M. F. Alhamid and M. Guizani, "Autonomous monitoring in the healthcare environment: Reward-based energy charging mechanism for IoMT wireless sensing nodes," *Future Generation Computer Systems*, vol. 98, no. 6, pp. 565–576, 2019.
- [16] 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. on Knowledge Discovery and Data Mining*, Halifax, Canada, pp. 787–795, 2017.
- [17] C. Nicholas, 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," arXiv preprint arXiv:1701.08680, 2017.
- [18] I. Azimi, J. Takalo Mattila, A. Anzanpour, A. M. Rahmani and J. P. Soininen, "Empowering healthcare IoT systems with hierarchical edge-based deep learning," in *Proc. IEEE/ACM Int. Conf. on Connected Health: Applications, Systems and Engineering Technologies*, Chennai, India, pp. 63–68, 2018.
- [19] L. Li, K. Ota and M. Dong, "Deep learning for smart industry: Efficient manufacture inspection system with fog computing," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4665–4673, 2018.
- [20] R. Mahmud, F. L. Koch and R. Buyya, "Cloud fog interoperability in IoT-enabled healthcare solutions," in *Proc. 19th Int. Conf. on Distributed Computing and Networking*, Varanasi, India, pp. 1–10, 2018.

- [21] B. Rabindra, R. Priyadarshini, H. Dubey, V. Kumar and K. Mankodiya, "FogLearn: Leveraging fog based machine learning for smart system big data analytics," *International Journal of Fog Computing*, vol. 1, no. 1, pp. 15–34, 2018.
- [22] S. Tuli, R. Mahmud, S. Tuli and R. Buyya, "Fogbus: A blockchain based lightweight framework for edge and fog computing," *Journal of Systems and Software*, vol. 154, no. 4, pp. 22–36, 2019.
- [23] C. Vecchiola, X. Chu and R. Buyya, "Aneka: A software platform for .net-based cloud computing," *High Speed and Large Scale Scientific Computing*, vol. 18, no. 3, pp. 267–295, 2009.
- [24] V. Kandasamy, P. Trojovský, F. Machot, K. Kyamakya, N. Bacanin *et al.*, "Sentimental analysis of COVID-19 related messages in social networks by involving an N-gram stacked autoencoder integrated into an ensemble learning scheme," *Sensors*, vol. 21, no. 22, pp. 7582, 2021.
- [25] M. Abouhawwash, "Hybrid evolutionary multi-objective optimization algorithm for helping multi-criterion decision-makers," *International Journal of Management Science and Engineering Management*, vol. 16, no. 2, pp. 94–106, 2021.
- [26] A. Nayyar, S. Tanwar and M. Abouhawwash, *Emergence of the Cyber-Physical System and IoT in Smart Automation and Robotics: Computer Engineering in Automation*, Springer, Verlag GmbH, Heidelberg, Germany, 2021.
- [27] A. Garg, A. Parashar, D. Barman, S. Jain, D. Singhal *et al.*, "Autism spectrum disorder prediction by an explainable deep learning approach," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1459–1471, 2022.
- [28] A. H. El-Bassiouny, M. Abouhawwash and H. S. Shahan, "New generalized extreme value distribution and it's bivariate extension," *International Journal of Computer Applications*, vol. 173, no. 3, pp. 1–10, 2017.
- [29] A. H. El-Bassiouny, M. Abouhawwash and H. S. Shahan, "Inverted exponentiated gamma and it's bivariate extension," *International Journal of Computer Application*, vol. 3, no. 8, pp. 13–39, 2018.
- [30] M. Abouhawwash and M. A. Jameel, "KKT proximity measure versus augmented achievement scalarization function," *International Journal of Computer Applications*, vol. 182, no. 24, pp. 1–7, 2018.
- [31] H. S. Shahan, A. H. Bassiouny and M. Abouhawwash, "Bivariate exponentiated modified weibull distribution," *Journal of Statistics Applications & Probability*, vol. 8, no. 1, pp. 27–39, 2019.
- [32] M. Abouhawwash and M. A. Jameel, "Evolutionary multi-objective optimization using benson's karush-kuhn-tucker proximity measure," in *Int. Conf. on Evolutionary Multi-Criterion Optimization*, East Lansing, Michigan, USA, pp. 27–38, 2019.
- [33] M. Abouhawwash, M. A. Jameel and K. Deb, "A smooth proximity measure for optimality in multi-objective optimization using benson's method," *Computers & Operations Research*, vol. 117, no. 2, pp. 104900, 2020.
- [34] M. Masud, G. Gaba, K. Choudhary, M. H. ossain, M. Alhamid *et al.*, "Lightweight and anonymity preserving user authentication scheme for IoT-based healthcare," *IEEE Internet of Things Journal*, vol. 24, no. 2, pp. 1–12, 2021.
- [35] A. Ali, H. A. Rahim, J. Ali, M. F. Pasha, M. Masud *et al.*, "A novel secure blockchain framework for accessing electronic health records using multiple certificate authority," *Applied Sciences*, vol. 11, no. 21, pp. 1–14, 2021.
- [36] M. Abouhawwash, K. Deb and A. Alessio, "Exploration of multi-objective optimization with genetic algorithms for PET image reconstruction," *Journal of Nuclear Medicine*, vol. 61, no. 3, pp. 572–572, 2020.
- [37] M. Masud, G. S. Gaba, K. Choudhary, R. Alroobaea and M. S. Hossain, "A robust and lightweight secure access scheme for cloud-based e-healthcare services," *Peer-to-peer Networking and Applications*, vol. 14, no. 3, pp. 3043–3057, 2021.
- [38] M. Masud, G. Gaba, S. Alqahtani, G. Muhammad, B. Gupta *et al.*, "A lightweight and robust secure key establishment protocol for internet of medical things in COVID-19 patients care," *IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15694–15703, 2021.
- [39] H. Sun and R. Grishman, "Lexicalized dependency paths based supervised learning for relation extraction," *Computer Systems Science and Engineering*, vol. 43, no. 3, pp. 861–870, 2022.
- [40] H. Sun and R. Grishman, "Employing lexicalized dependency paths for active learning of relation extraction," *Intelligent Automation & Soft Computing*, vol. 34, no. 3, pp. 1415–1423, 2022.