A Smart Architecture Leveraging Fog Computing Fusion and Ensemble Learning for Prediction of Gestational Diabetes

Zeena N. Al-kateeb *, Dhuha Basheer Abdullah

College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq
Emails: zeenaalkateeb@uomosul.edu.iq; prof.dhuha_basheer @uomosul.edu.iq

Abstract

Gestational diabetes (GD) is a growing global concern, underscoring the need for early detection and effective management to prevent adverse health consequences. This paper presents an innovative and reliable architecture to predict gestational diabetes in pregnant women. While reducing the frequency of doctor visits by sending the necessary data via Internet of Things (IoT) technology and receiving the prediction results via a mobile application in real time. The proposed architecture is a fusion of fog computing hardware with ensemble machine learning to enable low-latency, energy-efficient solutions for data processing, and cloud computing. The GD_Fog architecture leverages fused fog computing and load balancing techniques to reduce latency, power consumption, Network bandwidth consumption, and response time, and cloud computing is used based on the concept of use on demand for more reliability while harnessing the power of group learning to improve prediction accuracy. In addition, GD_Fog can be configured for different operating modes to ensure optimal quality of service and prediction accuracy in various fog calculation scenarios, which meet different user requirements. Through extensive testing using real-world data from pregnant women, the framework shows promising results, outperforming the latest methods in accuracy and efficiency. Where the percentage of improvement in prediction accuracy was approximately 6.5% when using ensemble learning, and the improvement in energy use, amounted to approximately 87.01% when using fused fog computing instead of cloud computing. These results confirm the potential of the proposed structure as an invaluable tool for the early detection and effective management of gestational diabetes.

Keywords: Smart Architecture; Internet of Things; Fog Computing; Fusion; Ensemble Learning.

1. Introduction

Gestational diabetes is a form of diabetes that occurs during pregnancy and affects 2-10% of pregnant women globally, is usually diagnosed between 22 and 26 weeks of gestational [1]. This condition is caused by hormones produced during pregnancy, which can disrupt the normal functioning of insulin, a hormone responsible for regulating blood sugar levels. When a woman has gestational diabetes, her body is unable to produce enough insulin to keep blood sugar levels in check, resulting in elevated blood sugar levels. If left untreated, gestational diabetes can pose health risks to both the mother and the baby, including metabolic complications, high birth weight, premature birth, and delivery difficulties [2,3]. Treatment typically involves a healthy diet, physical activity, and in some cases, insulin therapy to manage blood sugar levels and ensure a safe pregnancy [4]. Monitoring the vital parameters of a pregnant woman is critical in predicting and preventing gestational diabetes. Regular monitoring helps healthcare providers identify women at risk early and provide timely treatment to prevent complications. Monitoring blood sugar levels can aid in determining the best treatment plan, which may include dietary changes, physical activity, and insulin therapy [5,6]. Early treatment of gestational diabetes reduces the risk of complications for

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

both the mother and the baby, such as high birth weight, premature birth, and delivery difficulties. Controlling gestational diabetes also leads to better maternal and fetal outcomes, including a healthier pregnancy and delivery and a lower risk of long-term health problems for the baby [7]. Therefore, monitoring the vital parameters of a pregnant woman is essential for predicting and preventing gestational diabetes and ensuring the best possible outcomes for both mother and child [8]. However, frequent visits to the doctor will drain a lot of time and effort, and pregnant women will bear more burdens. Hence, the integration of Artificial Intelligence (AI) and the Internet of Things (IoT) has emerged as a great solution in revolutionizing the healthcare industry, especially in the field of telemedicine. By combining the power of machine learning with a vast network of connected devices, healthcare providers can now accurately predict and diagnose diseases delivering greater speed than ever before [9,10]. Through artificial intelligence algorithms that analyze massive amounts of patient data collected through IoT sensors, healthcare professionals can quickly detect patterns and anomalies that may indicate the onset of disease. This enables early intervention, personalized treatment plans, and ultimately better patient outcomes. Moreover, the use of telemedicine through artificial intelligence and the Internet of Things has made health care more accessible to people living in remote areas or with limited mobility, enabling doctors to remotely monitor and diagnose patients [11,12,13,14]. This allows the identification of patterns and risk factors associated with various diseases, including gestational diabetes [15]. Currently, there are several studies underway to explore the use of AI in predicting gestational diabetes, with promising results. These studies aim to improve the accuracy and efficiency of gestational diabetes screening and to develop new AI algorithms for early detection and management of the condition [16]. These studies aim to provide better care for pregnant women and reduce the risks of health problems for both the mother and the child [17,18]. However, the time factor and the speed of response still have a major role in the success of the use of modernity in the integration of the Internet of Things and artificial intelligence in the field of telemedicine. After. From here, fog computing has emerged as a crucial layer for processing and analyzing data generated by IoT devices [19]. With the prevalence of IoT devices in healthcare, the volume of data generated can be overwhelming for traditional cloud computing systems to handle [20]. Fog computing, which distributes computing resources close to the edge of the network, can reduce this burden and improve the efficiency and speed of data processing. This is particularly important for telemedicine applications, where real-time analysis of patient data can be critical to providing timely and accurate medical care [21]. Fog computing is not intended to replace cloud computing, but instead serves as an extension of it by utilizing resources located close to the edge of the network [22], where sensitive data can be processed and analyzed locally rather than sent to a central cloud server. Fog computing has become a key component to enable the deployment of IoT telemedicine solutions, providing improved efficiency, privacy, and security for the benefit of both patients and healthcare providers [23]. In addition to the necessity of distributing the loads on the fog nodes to ensure a reduction in response time, which provides greater opportunities to ensure the success of such systems and architectures and to increase the acceptance of their use [21].

The integration of the Internet of Things (IoT) with machine learning (ML), fused fog computing, cloud computing, and load balancing has the potential to massively revolutionize healthcare by improving patient outcomes and enhancing clinical services, particularly in the field of predicting diabetes pregnancy. By taking advantage of IoT devices such as continuous glucose monitoring systems, cloud data storage, and processing, all of this enhances the ease and smoothness of using all these modern technologies combined by designing a mobile application that sends data and receives predictions to provide warnings to pregnant women everywhere, saving them the trouble. Going to the doctor, reducing high costs, and working to provide health care as a service.

This paper makes the following contributions:

• Innovative Architecture: We propose an innovative architecture that fused ensemble machine learning with fog computing. The architecture leverages the power of emerging technologies such as the Internet of Things (IoT) and mobile applications to provide real-time predictions.

• Improved Prediction Accuracy: By incorporating ensemble learning techniques, GD_Fog achieves an improvement in prediction accuracy.

• Energy-efficient Computing: Where utilizes fog computing, and the decentralized computing paradigm, to process data locally and reduce the energy consumption associated with cloud computing. Our framework achieves an energy use improvement of traditional cloud computing approaches.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

• Support for scalability and availability in case of hardware failure through the ability to add cloud devices and find replacement devices

• Latency and Response Time Reduction: The fused of fog computing , load balancing and cloud elasticity techniques significantly reduces latency, network bandwidth consumption, and response time.

• Configurability for Diverse Scenarios: GD_Fog offers configurability to adapt to different fog computing fusion scenarios and user requirements.The subsequent sections in the paper are organized as follows:

Section 2 provides a comprehensive literature survey, examining the existing body of knowledge in the field. Section 3 outlines the methodology and system architecture employed in this study, shedding light on the underlying framework. Section 4 delves into the implementation of GD_Fog, offering a detailed explanation of its design and functionality. Section 5 focuses on the dataset utilized in this research, elucidating its composition and relevance to the study objectives. Section 6 conducts a thorough performance evaluation, analyzing the effectiveness and efficiency of the proposed approach. Section 7 comprises an illustrative discussion, where the findings are interpreted and contextualized. This is then followed by a concluding section, summarizing the key insights and implications of the study.

2. Related Work

The literature contains numerous studies on the gestational diabetes prediction, machine and deep learning algorithms, and the use of fog computing to improve response time. However, these topics are not usually discussed in conjunction, and there are gaps and inadequacies regarding their reliability and availability. To address these issues, we present a sophisticated smart architecture that integrates a range of cutting-edge technologies. Specifically, our proposed architecture leverages IoT components to facilitate self-monitoring, vital sign measurements, and machine learning and group learning algorithms to enhance prediction accuracy for gestational diabetes. Moreover, we employ fog and cloud computing to optimize responsiveness and bolster reliability.
In 2020, Zhang et al.[24] conducted a study that developed a straightforward and economical mammogram for early prediction of gestational diabetes mellitus (GDM) risk in the first trimester. The study involved 1385 pregnant women, and the findings indicated a considerably greater likelihood of GDM in women aged ≥35 years compared to those aged <25 years. In the same year, [25] investigated a non-invasive system for detecting diabetes using IoT sensors and machine learning algorithms. The system also uses an ESP thirty-two Wi-Fi module that sends the data to the cloud internet, which can be accessed through a mobile application. The data collected from 100 patients are used to train and test a statistical regression classifier that maps breath acetone to glucose value. The results show that the system is effective in detecting diabetes providing a promising path for developing non-invasive [25].
Xiong et al. [26] presented a study aimed to develop a predictive model for gestational diabetes mellitus (GDM). A total of 490 pregnant women was included, and a support vector machine and light gradient boosting machine were used to estimate associations and build the model. The study suggests that PAT-PT and PAT-APTT could be potential novel biomarkers for the earlier diagnosis of GDM. In 2021, Amarnath et al. [27] conducted a study that compared different classifiers to predict gestational diabetes mellitus (GDM) and found that the RF classifier with tuned parameters had the highest accuracy rate. They also found that using the GBM method and min-max normalization techniques in data preprocessing improved the data quality. The study suggested using an internally combined approach due to correlated attribute values in the GDM dataset, and the RF algorithm was found to be effective in predicting GDM occurrence.
In Ref. [28] proposed a real time IoT-based remote monitoring system for pregnant women in remote areas to reduce mortality rates. The authors collected a dataset of 10,000 pregnant women and applied machine learning techniques to predict mortality rates based on factors such as the mother's age and frequency of pregnancy. The authors tested various classification algorithms and found that the two class SVM model had the most accurate prediction performance. In 2022, Zhang et al. [29] present a study about meta-analysis examining published prognostic models for predicting the risk of gestational diabetes mellitus (GDM) using machine learning (ML) techniques. The study identified predictors applicable to the models and compared the performance of different ML methods. The authors analyzed 25 studies that included women over 18 years of age without a history of viral disease. Non-logistic regression models performed better than logistic regression

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

models. Another paper [30] proposes an IoT-based self-care system that utilizes machine learning algorithms to predict diabetes mellitus early. The system combines both bagging and boosting methods to create a hybrid ensemble model for classification. The results of the study suggest that blood sugar levels are the most significant predictor of diabetes mellitus. In Ref. [31] A comprehensive framework for monitoring pregnant women using an IoT-Fog-based healthcare system had suggested, consisting of three layers: IoT layer, Fog layer, and Cloud layer. The main contribution is at the fog layer producing the GDM module to implement two influential tasks: Data Finding Methodology (DFM) and Explainable Prediction Algorithm (EPM) using DNN. The DFM is used to replace unused data to free up cache space for new incoming data items, and the EPM predicts the occurrence of GDM in the second trimester of pregnancy using vital signs, laboratory tests, and patient demographics.
Nancy et al. [32] introduced a research initiative proposing an IoT-Cloud-based smart healthcare system for heart disease risk prediction using a fuzzy inference system (FIS) and bidirectional LSTM recurrent neural network. This system has the potential to provide personalized recommendations for individuals based on their health condition and heart specialist advice. In 2023, a new framework IoMT (Internet of Medical Things) is introduced for diabetes prediction [33], utilizing WBANs, cloud computing, fog computing, fuzzy logic, and machine learning algorithms (SVM, RF, and ANN). The accuracy performance of the fuzzy logic in fog and machine learning algorithms in the cloud is presented, with SVM having the highest accuracy at 89.5%. The IoMT approach is built using the CSMA-CA-based IEEE 802.15.6 protocol and the AODV routing algorithm, with examined end-to-end delay and throughput results of heterogeneous nodes.
Hennebelle et al.[34] explore a machine learning-based smart healthcare framework called HealthEdge for the prediction of type 2 diabetes in an integrated IoT-edge-cloud computing system. The system analyses diabetes risk factors using medical sensors/devices and predicts the incidence of diabetes in an individual. The collected data is sent to edge servers for preprocessing and transformation, then to the cloud for machine learning model development. The developed model is then used by edge servers for predicting diabetes.
Table.1 presents a comparative analysis between the proposed system and related works in this field.

Table 1: Comparison Analysis between The Proposed System and Related Works.

Ref.	**Year**	**Disease**	**Multi_ Environment**	**IoT**	**Cloud**	**Fog**	**Real Time**	**Scalable**	**Reliability**	**Ensemble Learning**
[23]	2020	gestational diabetes								
[24]	2020	diabetes	👍	👍						
[25]	2020	gestational diabetes	👍	👍			👍			
[26]	2021	gestational diabetes	👍						👍	👍
[27]	2021	monitor pregnant women	👍	👍	👍		👍			
[29]	2022	Diabetes	👍	👍			👍			👍
[30]	2022	gestational diabetes	👍	👍	👍	👍	👍	👍		
[31]	2022	heart disease	👍	👍	👍		👍			
[32]	2023	diabetes	👍	👍	👍	👍	👍			
[33]	2023	diabetes	👍	👍	👍		👍			
This work		gestational diabetes	👍	👍	👍	👍	👍	👍	👍	👍

Based on the information provided in Table 1, it is evident that there are certain challenges that need to be addressed in order to fully harness the potential of IoT-based fog computing fusion in healthcare systems. These challenges include:

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

- Development of an efficient IoT-based healthcare application capable of processing a large volume of Gestational diabetes data while minimizing energy consumption and ensuring low response time.
- Implementation of a well-organized resource scheduling technique specifically designed for fog computing environments. This technique should enable optimal utilization of resources to execute user workloads, ensuring that deadlines are met effectively.
- Integration of an ensemble machine learning-based fused fog computing model that can automatically diagnose Gestational diabetes in patients in real time. This model should leverage the power of ensemble machine-learning algorithms to enhance the accuracy and efficiency of the diagnostic process.

By addressing these challenges, healthcare systems can fully capitalize on the capabilities of IoT-based fog computing fusion, enabling efficient processing of Gestational diabetes data, resource optimization, and real-time diagnosis for improved health care outcomes.

3. Methodology

Our architecture adopts the principle of FogBus which is a comprehensive framework that enables the creation and deployment of integrated Fog-Cloud environments [35]. Where it offers a platform-independent application execution and structured communication, allowing for the connection of health-related IoT sensors through gateways [36]. The platform offers a range of features, including real-time data processing, edge analytics, and machine learning capabilities. The FogBus platform also provides a set of tools for building and deploying IoT applications, including device management, data integration, and security features. Tasks are initiated in the edge, then fog broker nodes and resources are efficiently managed. To enhance the reliability and robustness of the fog environment [37,38].

3.1. System Architecture

The GD_Fog model is a healthcare solution that leverages IoT and fog-enabled cloud computing to effectively manage data related to gestational diabetes patients and diagnose their health status to determine the severity of the condition. By integrating a variety of hardware instruments through software components, GD_Fog enables smooth and seamless integration of Edge-Fog-Cloud for quick and accurate results delivery. The GD_Fog model can work in two different phases, one of which focuses on load balancing and using cloud r elasticity to reduce response time, achieve reliability, and avoid failure through fault tolerance. The second phase works to improve the accuracy of prediction by operating all workers' nodes for training and testing purposes, based on the principle of ensemble learning work. Fig.1 shows the architecture of GD_Fog, which consists of various hardware and software components, which will be further explained in detail.

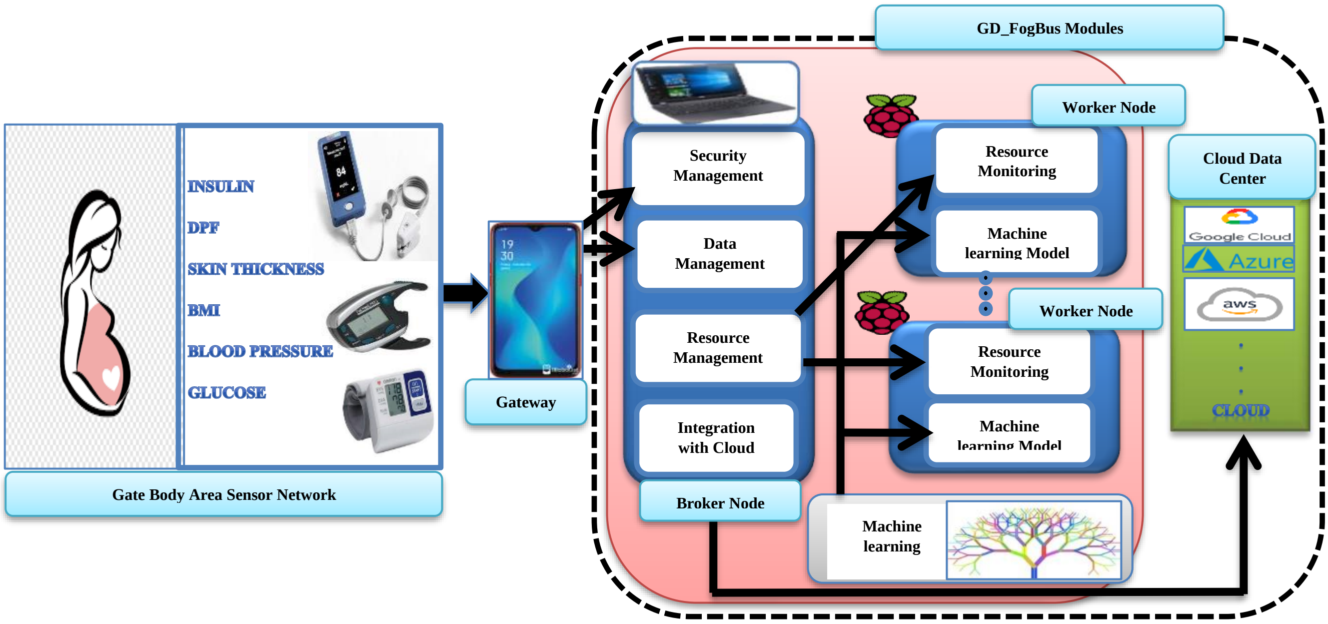


Figure 1: The GD_Fog Architecture

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

3.2. Hardware Components of GD_Fog

The hardware components of the GD_Fog model consist of:

1. Body Area Sensor Network: This component comprises various types of sensors that collect several Medical Variables (Independent) data, including glucose level, blood pressure, insulin level, BMI, and the Diabetes Pedigree Function (DPF), as well as the number of times the woman has been pregnant, and the age of the pregnant woman. The collected data from the pregnant woman is then transmitted to the connected gateway devices.
2. Gateway Devices: There are three types of gateway devices - mobile phones, laptops, and tablets - that serve as fog devices to gather data from various sensors and transmit it to the Broker/Worker nodes for additional processing.
3. GD_FogBus Modules: The GD_FogBus framework consists of the following components:

(a) Broker Node: This component receives job requests and data inputs from the Gateway devices. The Request Input Module receives job requests from the Gateway devices, while the Security Management Module ensures secure communication between components and protects the collected data from unauthorized access or malicious tampering, thus improving system credibility and data integrity. The Arbitration Module, which is part of the Resource Manager in the Broker Node, processes load statistics from all worker nodes to decide which node or subset of nodes to allocate tasks to in real-time.

(b) The Worker Node: performs tasks assigned by the Broker node's Resource Manager. It may consist of embedded devices and Single Board Computers like Raspberry Pis. The Worker node can also incorporate advanced deep learning models for data processing and analysis, along with other components for data processing, filtering, and mining, big data analytics, and storage. The input data is directly received from the Gateway devices and results are generated and shared with the same. In some cases, the Broker node can also function as a Worker node in the GD_Fog model.

(c) The Cloud Data Center: In situations where the fog infrastructure is overwhelmed, the GD_Fog system leverages the resources of Cloud Data Centers (CDC) to handle latency-tolerant services or large amounts of input data that exceed the average size. This enhances the robustness and speed of the system and enables data processing to be location-independent.

3.3. Software Components of GD_Fog

The components of the GD_Fog model include the following software components:

(a) Data Pre-Processing: The pre-processing stage is critical for the effective utilization of data in machine learning classification algorithms. Data preparation begins with filtering and processing the input data. This involves utilizing data analytics tools for cleaning and refining the data, such as removing duplicates, identifying and correcting null values, and replacing abnormal values. For instance, in medical records, it's not possible for certain readings, such as blood pressure or glucose levels, to have a value of zero. In such cases, the mean value for that column is substituted. On the other hand, Skin Thickness, Insulin, and BMI have skewed distributions, making the median a better choice as it is less impacted by outliers.

With the processed data, the Set Partitioning In Hierarchical Trees (SPIHT) algorithm is utilized for Principal Component Analysis (PCA), while applying the Singular Value Decomposition (SVD) technique to encrypt the data [39]. The objective is to extract the crucial components from the data feature vectors that have an impact on the health status of patients the system utilizes machine learning, based on continuous training data from health care providers and doctors, to make informed recommendations for medication and check-up [40]. The results are stored in a database for future refinement and training.

(b) The Resource Manager: is made up of two key components, the Workload Manager and the Arbitration Module. The Workload Manager is responsible for managing job requests and queues for data processing tasks, as well as handling large amounts of data. The Arbitration Module, located within the Broker node, schedules the available fog or cloud resources to process the tasks that have been queued and managed by the Workload Manager. It determines which device (the Broker, the Fog Worker node, or the Cloud Data Center) should receive the data for processing to achieve optimal performance by balancing the workload among the various devices. GD_Fog gives users the flexibility to set their own load balancing and arbitration strategies based on their application needs.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

(c) The Machine Learning Module: This module is responsible for training (RandomForestClassifier) machine learning using the dataset to classify feature vectors, which are obtained from pre-processing the data from the Body Area Sensor Network. Upon receiving a task from the Resource Manager, this module also predicts and produces results for the data received from the Gateway devices.

The data was split into training, validation, and testing sets with a ratio of 80:20. The random_state parameter was set to 0 to ensure that the same random split is generated every time the code is run. The training set was used to train the model, the validation set was used to fine-tune the model, and the test set was used to evaluate how the model performed on new data. The trained model can be stored in all nodes that can process it, either by storing it in a common database or by training separate models distributed across different nodes. During diagnosis, a patient's data is input into the model, which makes a forward pass on the machine learning and outputs a prediction of whether the pregnant woman has Gestational Diabetes or not. The ensemble method of Bagging combines the results of various models to provide more accurate results. The worker that receives the input data multicasts it to other worker nodes and the predicted results of each worker node are sent back to the worker assigned to this task. The majority of prediction is obtained through bagging and is sent to the gateway device. While ensemble learning gives better accuracies, it also has higher response time and network overhead, which can be disabled if latency is a critical concern. The next sections provide further details on these findings

(d) The Ensemble Module: it is located within the Broker node and is responsible for collecting the predicted results from multiple models and determining the output class, indicating whether or not the pregnant woman has Gestational diabetes, through voting. This node is tasked with both distributing data and gathering results from other worker nodes.

For our application, we employed an ensemble of machine learning (RandomForestClassifier) as a model to conduct predictive analysis for binary classification problems. Firstly, the model was trained on the Pima Indians Diabetes Database's Gestational Diabetes data along with the corresponding known output class. After the training process, the model was utilized to predict real-time data input results, as illustrated in Fig. 2.

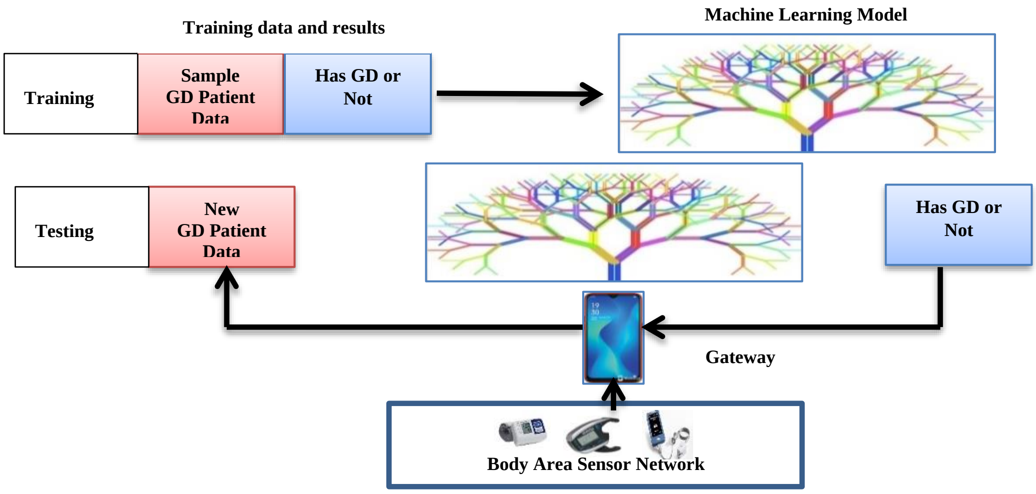


Figure 2: Training and Testing

3.4. GD_Fog Topology

The GD_Fog components need to share significant amounts of data, information, and control signals among themselves, thus stable network communication is crucial. To ensure this, the components are organized in a topology shown in Fig. 1. Communication across all Edge devices is facilitated by the FogBus [41,10].

The Broker Node (Master) controls the Worker Nodes as (Slaves). All the Edge devices, including the Gateway devices, Broker node, and Worker nodes, are present in the same Local Area Network (LAN). The Resource Manager software component resides in the Broker Node, so the Gateway

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

devices send job requests to it. The arbitration results are then received by the Gateway device, which instructs it where to send the data.There are three possible scenarios:

- The Broker node processes data as a Worker Node.
- The Worker node processes the transmitted data.
- The Cloud Data Center processes the transmitted data.

Depending on the scenario, the Gateway device may send the data directly to the Worker node or Broker node (with/without cloud forwarding). The Broker may provide computation services only when it has sufficient resources or if the worker nodes are overloaded. If the data needs to be forwarded to the Cloud, it goes through the Broker node, as the Gateway may not have access to the Virtual Private Network (VPN) in which the Cloud Virtual Machine resides Fig. 3 illustrates the communication sequence in GD_Fog.

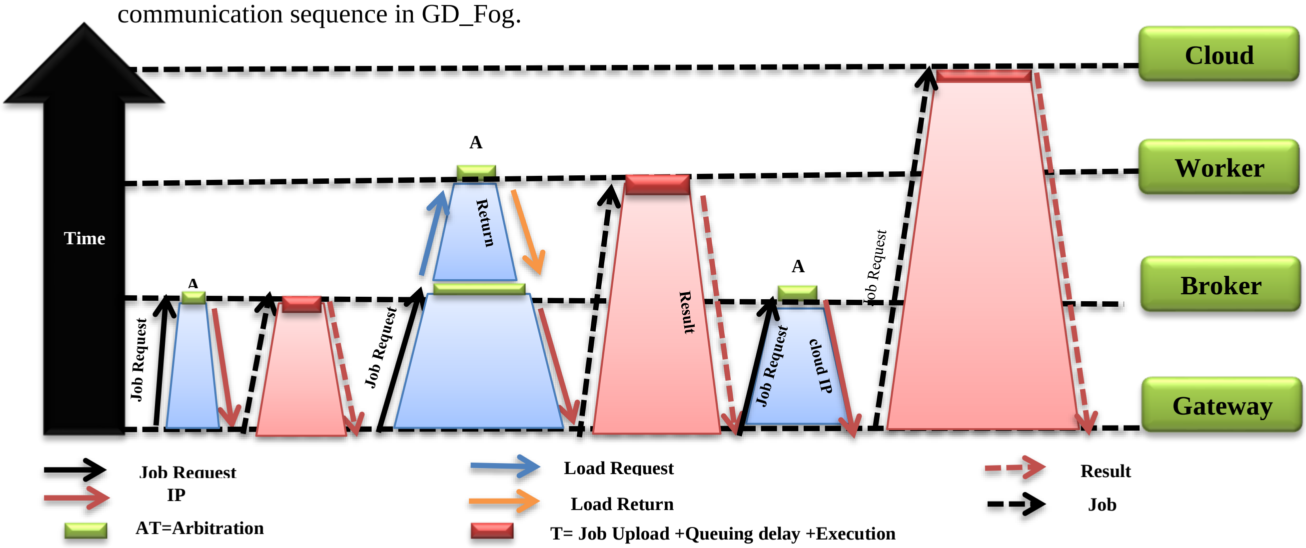


Figure 3: Communication sequence in GD_Fog

3.5. The Communication Sequence

The GD_Fog system uses predefined protocols for communication between its hardware components, as outlined in Fig. 4. When a Gateway device needs to send a job request, it first contacts the Broker node. Depending on the scenario (Broker Only, Worker Node, or Cloud), the Broker node will either provide the IP address of the least loaded Worker node or its IP address, with or without cloud forwarding.

If the workers are all heavily loaded or compromised, the Broker node will handle the job itself. If there are some workers with lighter loads, the Broker node will send the Gateway device the IP address of the least loaded worker. The Gateway device then sends the job data to the Worker node or the Broker node, which will perform pre-processing and prediction before sending the results back.

In the case of cloud forwarding, the Broker node acts as a relay, forwarding the data to the Cloud Data Center (CDC) on behalf of the Gateway device Fig. 4 describes the load balancing and cloud elasticity as a flowchart. This provides an extra layer of security as the IoT sensors and Gateway devices are only connected to the LAN, not the internet. Although the CDC has more resources available, processing the job at the CDC will result in higher latency due to communication overhead and delays at both the Broker node and the CDC.

When the ensemble is enabled, the data received by the Broker node or Worker node is forwarded to all other Edge nodes and the majority class is determined through bagging.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

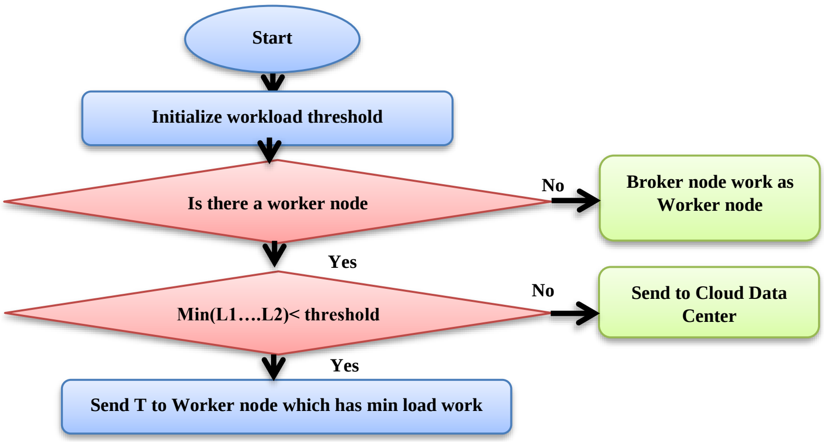
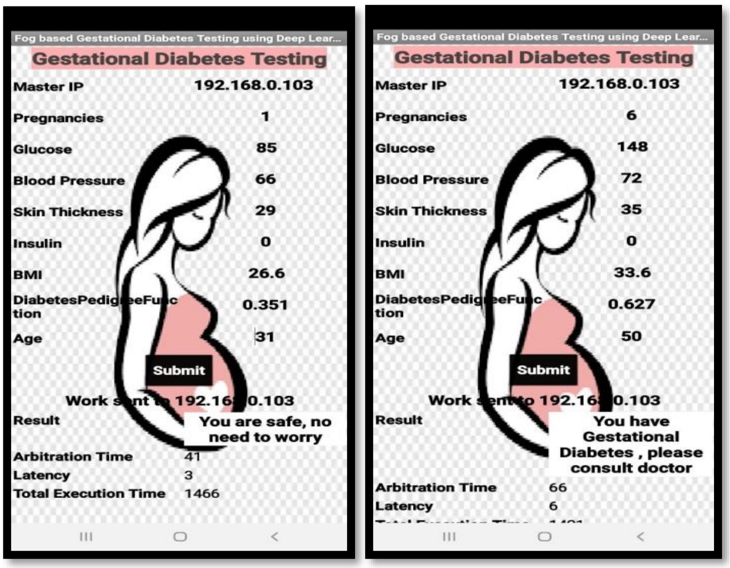


Figure 4: Load Balancing and Cloud Elasticity Flowchart.

3.6. Communication and Android interface

The Gateway device utilized an Android program called Gestational Diabetes Test to transmit information to the Broker/Worker nodes. Fig.5 shows the user interface of this application, which allows the Gateway to serve as a mediator between the Body Sensor Network and the Worker nodes. Communication is facilitated using HTTP RESTful APIs, with input data being uploaded and the results downloaded from the Gateway device via HTTP POST. Each Worker node, the Broker node, and CDC possess pre-trained machine learning models and pre-processing software.



A B

Figure 5: The User Interface of The GD_Fog Application
A: Negative Example, B: Positive Example

4. GD_Fog Implementation

Section 5 describes the various components that were implemented using different programming languages. The pre-processing and ensemble machine learning components were implemented in Python, where the pre-processing module normalizes the dataset based on the median or mean values of the field parameters and their distribution. The ensemble machine learning component used the SciKit learn Library and the Random Forest Classifier to implement a voting scheme. The base classifier used was a machine learning model as described in Table 2, and the model randomly distributed the data among the classifiers to train them. At diagnosis time, the model takes all predicted classes as input and outputs the majority of prediction.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

Table 2: The Base Classifier Model

Size of input layer	8	Number of features of the data
Size of output layer	2	Binary classification has Gestational Diabetes or not
N_Estimators	6	
Criterion	Entropy	Means that the entropy criterion is used
Max_depth	Default	Less than Min_Samples_split samples.
Min_Samples_split	2	Default
Min_Samples_leaf	1	Default
Max_Features		Default

The Android application was developed using MIT's App Inventor1 and saved the data attributes in a .csv file, which was uploaded to the broker node using HTTP POST. The broker node had an Arbitration Module that selected the worker node with the minimum CPU load for task execution. The Execution Interface Module in each worker received the data and instantiated the Ensemble machine learning model code for analysis. The result was assembled using the bagging strategy and sent back to the gateway device (anandroid application).

5. Used Dataset

The dataset used in this study was the Pima Indians Diabetes Database [2], which was obtained from the National Institute of Diabetes and Digestive and Kidney Diseases. The dataset includes diagnostic measurements for predicting diabetes in patients. The patients included in the dataset are all females of at least 21 years old, and they are of Pima Indian heritage. This was done using a binary classification system, where a value of 0 indicates that the woman does not have Gestational Diabetes and a value of 1 indicates that she does. We analyzed 9 important attributes of the data to determine the pregnant woman's health status. The Pima Indian Diabetes dataset includes the 9 attributes ,Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function (DPF), Age, Outcome which it a class variable that takes the value 0 or 1, where 1 indicates the individual has diabetes and 0 indicates they do not. Of the 768 instances in the dataset, 268 have a value of 1 for the Outcome variable, while the remaining 500 have a value of 0. Table 3 describes the details of only 10 pregnant women.

Table 3: The Details of The First 10 Pregnant Women from the Pima Indian Dataset

Pregnancies	Glucose	BloodPressure	Skin Thickness	Insulin	BMI	DPF	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0
3	78	50	32	88	31	0.248	26	1
10	115	0	0	0	35.3	0.134	29	0
2	197	70	45	543	30.5	0.158	53	1
8	125	96	0	0	0	0.232	54	1

6. The Performance Evaluation

To prove that the GD_Fog model is viable and effective, we put it into action on real devices using the FogBus framework. We applied the model to detect Gestational Diabetes in patients using machine learning in a Fog computing environment. We assessed its accuracy and response time, as well as its network and energy consumption, and found that the GD_Fog model is both efficient and has minimal overhead.

6.1. **The Experimental Setup**: The following is a description of the experimental setup and hardware configurations used in the evaluation of GD_Fog:

- Gateway Device: Samsung Galaxy S9 with Android 9

- Broker/Master Node: Lenovo with Intel(R) Core (TM) i7-7500U CPU @ 2.70GHz 2.90 GHz, RAM 7.90, and 64-bit Windows 10, using Apache HTTP Server 2.4.34 for deployment.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

- Worker Node: Raspberry Pi 4 Model B Rev 1.2 with ARM Cortex-A53 quad-core SoC CPU @ 1.4 GHz, 1 GB LPDDR2 SDRAM, IEEE 802.11 Wifi, and Raspbian Stretch Operating System with Apache HTTP Server 2.4.34.

- Public Cloud: Microsoft Azure B1s Machine, with 2vCPU, 12 GB RAM, 300 GB HDD, and Windows 10 Pro.

During the experiments, the Microsoft Performance Monitor was used to record data parameters at the Master and the Azure VM, while the Raspberry Pi circuits used NMON Performance Monitor. Microsoft Network Monitor 3.4 was used at the Broker node to measure network bandwidth consumption, and the vnStat tool was used in the Raspberry Pis.

6.2. Experiments on The Characteristics of an Architecture:

The GD_Fog architecture was assessed through experiments that focused on its characteristics. The dataset was divided into two parts: 80% for training and 20% for testing. To evaluate the performance of the model, various characteristics were analyzed: accuracy of prediction, time characteristics, network bandwidth usage, and power consumption. Accuracy was defined as the percentage of correctly predicted pregnant women with gestational diabetes by the model. Time characteristics included arbitration time, latency, execution time, and jitter. Network bandwidth usage was compared across different fog scenarios. Power consumption was also analysed.

6.2.1. The Prediction Accuracy

The increase in the number of Worker nodes causes a gradual rise in training accuracy since each node learns a model for the data received by it. As the number of nodes increases, each node receives few examples, and therefore training the model for multiple epochs over fits the samples, leading to an increase in training accuracy. Fig. 6 displays how the training accuracy changes with the number of Edge nodes.

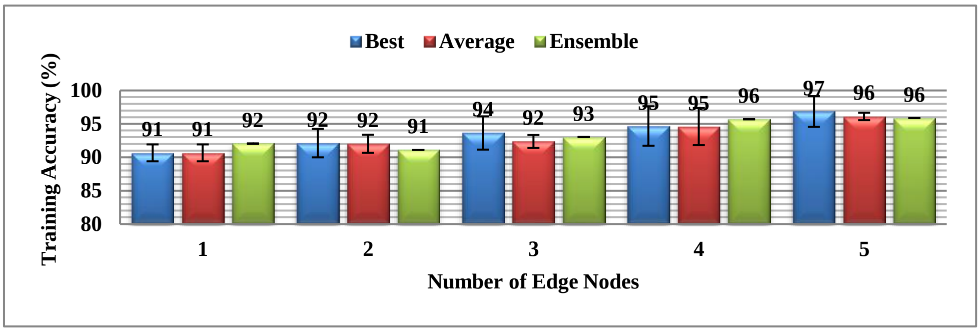


Figure 6: The Training Accuracy with Number of Edge Nodes.

The test accuracy declines with a higher number of nodes since each node receives a smaller subset of the training data, making it unable to generalize the model. It is also noteworthy that ensemble learning always yields significantly higher accuracy than the case without an ensemble. Fig. 7 illustrates the changes in test data accuracy as the number of Edge nodes increase.

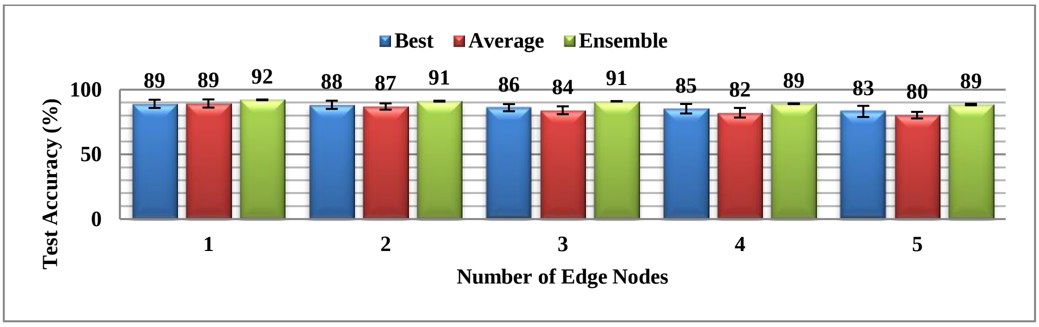


Figure 7: Test accuracy with the number of edge nodes.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

6.2.2. The Prediction Confidence

The machine learning model generates two probabilities (p0 and p1) when predicting whether a pregnant woman has Gestational Diabetes or not, where p0 represents the probability of no Gestational Diabetes and p1 represents the probability of Gestational Diabetes. These probabilities add up to 1. The model's confidence measure for a prediction (p0, p1) is determined by 100 × (2 × max (p0, p1) - 1), which gives a range of values between 0 and 100. For example, if the predicted probabilities are both 0.5, the confidence is 0. If the probabilities are between 0.9 and 0.1, the predicted class is 0% and the confidence is 80%.

The confidence of the binary classifier for the complete test dataset is shown in Fig. 8, where it is broken down into the subset of correctly predicted cases and the subset of incorrectly predicted cases.

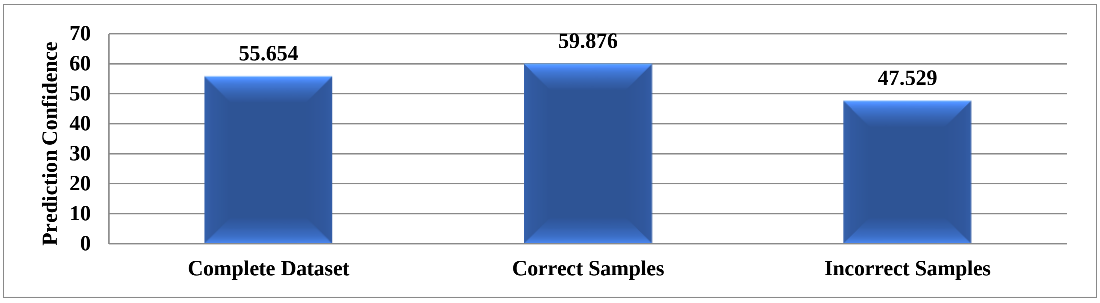


Figure 8: The Prediction Confidence of the Model for Different Subsets

If the confidence is less than 50%, the model recommends that the pregnant woman consult with a doctor as the prediction may not be reliable.

6.2.3. The Timing Characteristics

The arbitration time is the time it takes for the Broker node to determine which worker node should receive a task. If the task is sent directly to the Broker or Cloud, the arbitration time is negligible. When data is sent to worker nodes for ensemble learning, the arbitration time is similar to the non-ensemble case because the majority class choice is made by one of the worker nodes. As shown in Fig. 9.

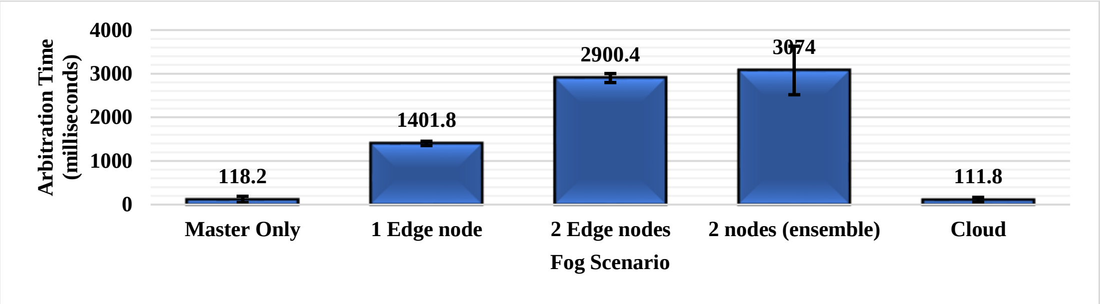


Figure 9: The Arbitration Time in Different Cases

Latency is nearly the same tasks sent to the Broker or any edge node, as all communication is through single-hop data transfers. The latency is slightly higher for ensemble learning, while it is very high for the Cloud setup because of the multi-hop transfer of data outside the LAN. These are shown in Fig.10

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

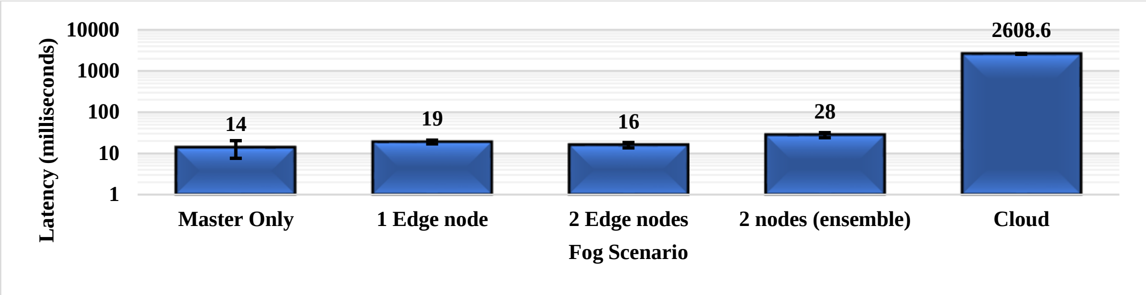


Figure 10: Latency in different cases.

Jitter, which is the variation of response time for consecutive job requests, is higher for the Broker-only case because the Broker performs. Jitter also slightly increases with the number of worker nodes. Fig. 11 shows the variety in jitter.

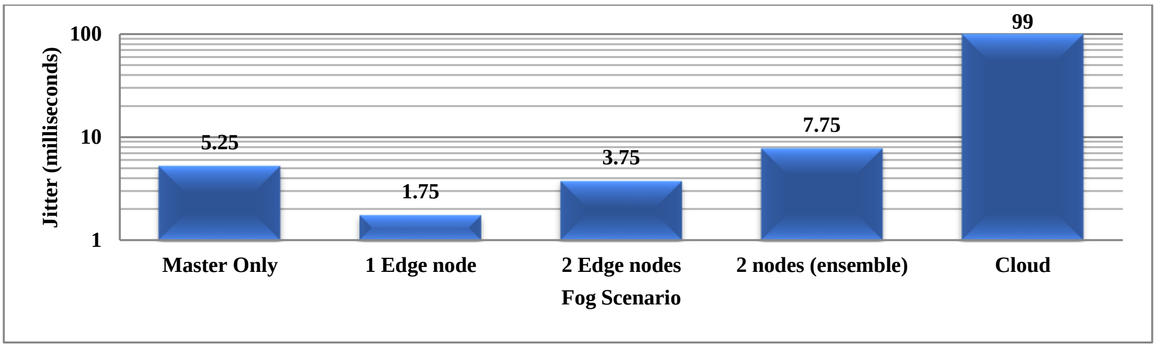


Figure 11: The jitter in different cases.

Execution time is lower in the Cloud setup due to higher resource availability. The Broker's execution time is less than that worker nodes because the workers are Raspberry Pis with low clock frequency. However, the execution time increases with ensemble learning because the worker node now needs to determine the majority class among all predicted classes. That is shown in Fig. 12.

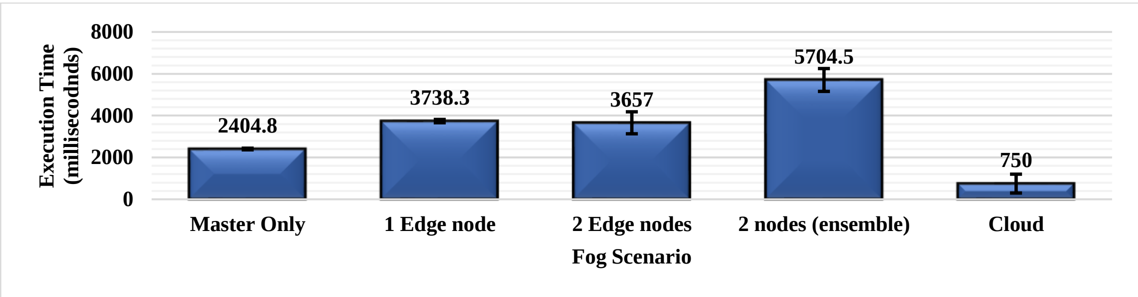


Figure 12: Execution time in different cases.

6.2.4. The Network Bandwidth Usage Characteristics

The usage pattern of network bandwidth is depicted in Fig. 13 for various scenarios. In the case where data is transmitted to all worker nodes, known as the ensemble case, the consumption of network bandwidth is at its peak.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

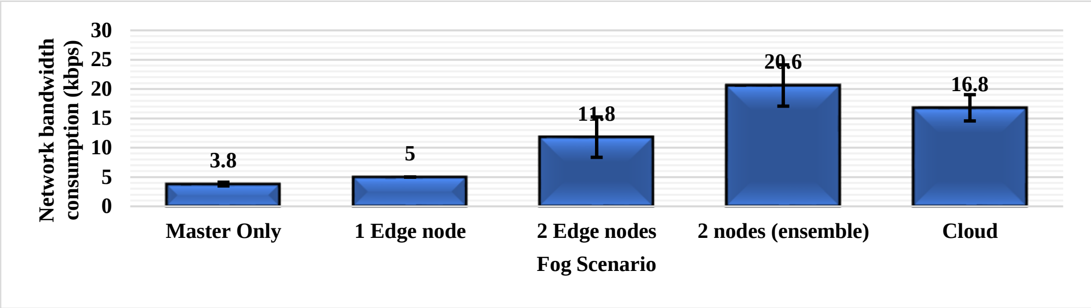


Figure 13: The network bandwidth used in different cases

6.2.5. The Power Characteristics

Experiments were performed to examine how much energy GD_Fog consumes in different situations. The outcomes are presented in Fig. 14 and demonstrate that the CDC consumes a lot more power than both the Broker node (laptop) and the Worker nodes (Raspberry Pi). Consequently, the power consumption is much greater in the Cloud setting when compared to the Edge setting. Additionally, increasing the number of Worker nodes leads to a corresponding increase in the power consumption of the GD_Fog framework.

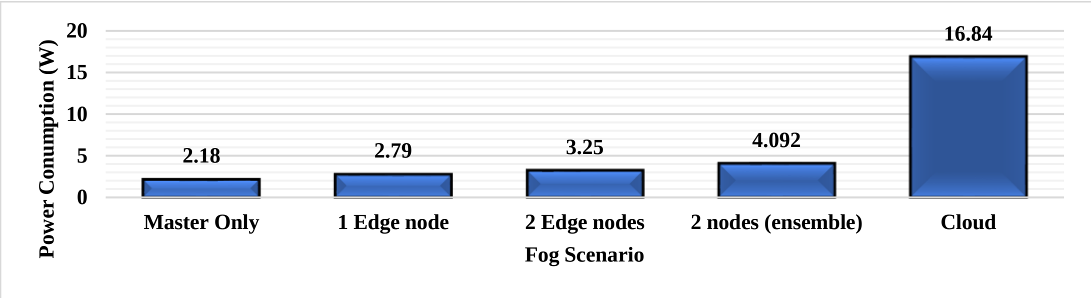


Figure 14: The power consumption in different cases.

Table 4 describes and shows a comparison of the results of the proposed architecture with some previous works.

Table 4: Comparison of the results of the proposed architecture with some previous works.

metric	Proposed Architecture			Ref [42]			Ref [43]		
		No. Node	Disease		No. Node	Disease		No. Node	Heart Disease
Training Accuracy	96%	5	Gestational Diabetes	87%	5	Heart Disease	86.67%	4	Heart Disease
Test Accuracy	89%	5	Gestational Diabetes	84%	5	Heart Disease	81%	4	Heart Disease
Arbitration Time	3074	2	Gestational Diabetes	3076.8	2	Heart Disease	1678.2	2	Heart Disease
Latency	16	2	Gestational Diabetes	17.4	2	Heart Disease	19.1	2	Heart Disease
Execution Time	3657	2	Gestational Diabetes	3660.9	2	Heart Disease	3122.4	2	Heart Disease
Network Bandwidth	20	2	Gestational Diabetes	20.6	2	Heart Disease	---		Heart Disease
Power Consumption	3.25	2	Gestational Diabetes	3.44	2	Heart Disease	5.07	2	Heart Disease

7. Discussion:

GD_Fog is an innovative and intelligent architecture that seamlessly combines Edge, Fog, and Cloud resources with user-friendly mobile applications, with a specific focus on developing a predictive application for gestational diabetes in pregnant women. This unique architecture prioritizes both speed and accuracy, setting it apart from previous computing models in healthcare applications within Fog Computing.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

Unlike other works in the field that neglect the utilization of edge resources, GD_Fog leverages advanced machine learning models to achieve highly accurate predictions of health characteristics. This distinction leads to superior disease detection accuracy, a critical factor in critical healthcare applications like gestational diabetes, where timely and precise results can significantly minimize harm to both the mother and fetus.

Furthermore, GD_Fog goes beyond existing machine learning techniques by harnessing the power of fog resource fusion for parallel computation and higher accuracy through ensemble methods. This approach provides even better results, as demonstrated by a notable improvement in prediction accuracy with the involvement of five edge nodes. This advancement surpasses what current systems can deliver.

The integration of the FogBus framework within GD_Fog offers tremendous flexibility. It allows users to configure the architecture according to their specific requirements, considering factors such as accuracy, response time, network, and power usage. This adaptability empowers users to customize the framework to suit their unique needs.

GD_Fog excels in simplicity and efficiency, utilizing fused fog computing, load balancing, and cloud elasticity to ensure the availability of resources and deliver rapid responses—an essential aspect in healthcare applications. The incorporation of ensemble learning further enhances prediction accuracy, while the intuitive mobile application streamlines data input and result retrieval for seamless user interaction.

The deployment system of GD_Fog caters to diverse user requirements, offering flexible configurations based on varying priorities of accuracy and speed. Whether low latency and minimal energy consumption are paramount, or improved prediction accuracy through ensemble bagging is desired, GD_Fog provides the necessary flexibility. Additionally, for computationally intensive tasks with acceptable latency, the CDC configuration enables successful execution resource-constrained edge worker nodes.

Overall, GD_Fog represents a significant advancement in healthcare architecture, combining simplicity, efficiency, and cutting-edge techniques to deliver accurate and timely results for gestational diabetes prediction. Its unique features and performance in various scenarios make it a noteworthy contribution to the field.

8. Conclusions

The provision of healthcare as a service is a massive undertaking. The focus of this research is solely on healthcare related to predicting gestational diabetes in pregnant women. The proposed solution is a new Fog-based Smart Healthcare System called GD_Fog, which leverages machine learning and IoT technologies to automatically predict gestational diabetes. GD_Fog functions as a fog service for healthcare, managing the data of pregnant women collected from various IoT devices. The system leverages the benefits of fog computing and load balancing to minimize latency and ensure swift response times. Additionally, it harnesses the cloud's elasticity to effectively handle sudden surges in demand, thereby ensuring scalability during emergency load spikes. The system integrates machine learning into Edge computing devices and applies it to real-life scenarios in gestational diabetes analysis. High-accuracy machine learning models require significant CPU and GPU resources for both training and prediction. However, this research has succeeded in embedding complex machine learning networks into Edge computing paradigms using novel communication and model distribution techniques such as ensemble, which has enabled high accuracy to be achieved with low latencies. The proposed system was validated using popular datasets and real-life pregnant women data analysis, and it provides real-time prediction results. The efficiency of the system was tested in terms of power consumption, network bandwidth, latency, jitter, training accuracy, testing accuracy, and execution time using the FogBus framework in a fused fog computing environment.

The advancement of health sensors and portable devices plays a crucial role in the accuracy and sophistication of smart health systems, presenting both opportunities and challenges. Ensuring swift and efficient communication is vital to uphold the quality of service in such systems.

For future endeavors, an area of focus lies in establishing an effective security mechanism by employing cutting-edge encryption algorithms. This will bolster the protection of sensitive medical information stored in fog and cloud databases, enhancing overall data security.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

Furthermore, integrating the proposed structure with big data frameworks holds promise for optimizing operations and improving efficiency. By leveraging the power of big data analytics, the system can extract valuable insights, leading to more effective decision-making and enhanced system performance.

Funding: “This research received no external funding”

Conflicts of Interest: “The authors declare no conflict of interest.”

References

[1] C. L. DeSisto, S. Y. Kim, and A. J. Sharma, “Prevalence estimates of gestational diabetes mellitus in the United States, pregnancy risk assessment monitoring system (PRAMS), 2007-2010,” *Prev. Chronic Dis.*, vol. 11, no. 12, pp. 1–9, 2014, doi: 10.5888/pcd11.130415.

[2] Y. Zhu and C. Zhang, “Prevalence of Gestational Diabetes and Risk of Progression to Type 2 Diabetes: a Global Perspective,” *Curr. Diab. Rep.*, vol. 16, no. 1, pp. 1–11, 2016, doi: 10.1007/s11892-015-0699-x.

[3] A. M. Egan, E. A. L. Enninga, L. Alrahmani, A. L. Weaver, M. P. Sarras, and R. Ruano, “Recurrent gestational diabetes mellitus: A narrative review and single-center experience,” *J. Clin. Med.*, vol. 10, no. 4, pp. 1–11, 2021, doi: 10.3390/jcm10040569.

[4] L. Guariguata, U. Linnenkamp, J. Beagley, D. R. Whiting, and N. H. Cho, “Global estimates of the prevalence of hyperglycaemia in pregnancy,” *Diabetes Res. Clin. Pract.*, vol. 103, no. 2, pp. 176–185, 2014, doi: 10.1016/j.diabres.2013.11.003.

[5] R. E. Ratner *et al.*, “Prevention of diabetes in women with a history of gestational diabetes: Effects of metformin and lifestyle interventions,” *J. Clin. Endocrinol. Metab.*, vol. 93, no. 12, pp. 4774–4779, 2008, doi: 10.1210/jc.2008-0772.

[6] J. B. Sussman, D. M. Kent, J. P. Nelson, and R. A. Hayward, “Improving diabetes prevention with benefit based tailored treatment: Risk based reanalysis of diabetes prevention program,” *BMJ*, vol. 350, no. February, pp. 1–10, 2015, doi: 10.1136/bmj.h454.

[7] V. R. Aroda *et al.*, “The effect of lifestyle intervention and metformin on preventing or delaying diabetes among women with and without gestational diabetes: The diabetes prevention program outcomes study 10-year follow-up,” *J. Clin. Endocrinol. Metab.*, vol. 100, no. 4, pp. 1646–1653, 2015, doi: 10.1210/jc.2014-3761.

[8] W. H. Herman *et al.*, “Impact of lifestyle and metformin interventions on the risk of progression to diabetes and regression to normal glucose regulation in overweight or obese people with impaired glucose regulation,” *Diabetes Care*, vol. 40, no. 12, pp. 1668–1677, 2017, doi: 10.2337/dc17-1116.

[9] Z. N. Aghdam, A. M. Rahmani, and M. Hosseinzadeh, “The Role of the Internet of Things in Healthcare: Future Trends and Challenges,” *Comput. Methods Programs Biomed.*, vol. 199, p. 105903, 2021, doi: 10.1016/j.cmpb.2020.105903.

[10] V. Thakare, G. Khire, and M. Kumbhar, “Artificial Intelligence (AI) and Internet of Things (IoT) in Healthcare: Opportunities and Challenges,” *ECS Trans.*, vol. 107, no. 1, pp. 7941–7951, Apr. 2022, doi: 10.1149/10701.7941ecst.

[11] E. Alreshidi, “Smart Sustainable Agriculture (SSA) solution underpinned by Internet of Things (IoT) and Artificial Intelligence (AI),” *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, pp. 93–102, 2019, doi: 10.14569/ijacsa.2019.0100513.

[12] D. B. Abdullah and Z. N. Al-Kateeb, “Prospects and Challenges of the Cloud of Things in Telemedicine,” in *2022 8th International Conference on Contemporary Information Technology and Mathematics (ICCITM)*, IEEE, Aug. 2022, pp. 1–7. doi: 10.1109/ICCITM56309.2022.10031829.

[13] Z. H. Arif and K. Cengiz, “Severity Classification for COVID-19 Infections based on Lasso-Logistic Regression Model,” *Int. J. Math. Stat. Comput. Sci.*, vol. 1, pp. 25–32, Apr. 2023, doi: 10.59543/ijmscs.v1i.7715.

[14] A. O. Salman and O. Geman, “Evaluating Three Machine Learning Classification Methods for Effective COVID-19 Diagnosis,” *Int. J. Math. Stat. Comput. Sci.*, vol. 1, pp. 1–14, Jan. 2023, doi: 10.59543/ijmscs.v1i.7693.

[15] K. Wang, Y. Zhao, and R. K. Gangadhari, “Analyzing the Adoption Challenges of the Internet of Things

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

(IoT) and Artificial Intelligence (AI) for Smart Cities in China," pp. 1–35, 2021.

[16] J. Shen, J. Chen, Z. Zheng, J. Zheng, and Z. Liu, "An Innovative Artificial Intelligence – Based App for the Diagnosis of Gestational Diabetes Mellitus (GDM-AI): Development Study Corresponding Author :," vol. 22, pp. 1–11, 2020, doi: 10.2196/21573.

[17] H. Shen, X. Liu, Y. Chen, B. He, and W. Cheng, "Associations of lipid levels during gestation with hypertensive disorders of pregnancy and gestational diabetes mellitus: A prospective longitudinal cohort study," *BMJ Open*, vol. 6, no. 12, 2016, doi: 10.1136/bmjopen-2016-013509.

[18] G. R. do Nascimento, M. do C. Borges, J. N. Figueiroa, L. V. Alves, and J. G. Alves, "Physical activity pattern in early pregnancy and gestational diabetes mellitus risk among low-income women: A prospective cross-sectional study," *SAGE Open Med.*, vol. 7, p. 205031211987592, 2019, doi: 10.1177/2050312119875922.

[19] K. Pareek, P. K. Tiwari, and V. Bhatnagar, "Fog Computing in Healthcare: A Review," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1099, no. 1, p. 012025, 2021, doi: 10.1088/1757-899x/1099/1/012025.

[20] M. Ijaz, G. Li, L. Lin, O. Cheikhrouhou, H. Hamam, and A. Noor, "Integration and applications of fog computing and cloud computing based on the internet of things for provision of healthcare services at home," *Electron.*, vol. 10, no. 9, 2021, doi: 10.3390/electronics10091077.

[21] H. J. de Moura Costa, C. A. da Costa, R. da Rosa Righi, and R. S. Antunes, "Fog computing in health: A systematic literature review," *Health Technol. (Berl).*, vol. 10, no. 5, pp. 1025–1044, Sep. 2020, doi: 10.1007/s12553-020-00431-8.

[22] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," *MCC'12 - Proc. 1st ACM Mob. Cloud Comput. Work.*, pp. 13–15, 2012, doi: 10.1145/2342509.2342513.

[23] A. Rejeb, K. Rejeb, H. Treiblmaier, A. Appolloni, S. Alghamdi, and Y. Alhasawi, *The Internet of Things (IoT) in Healthcare : Taking Stock and Moving Forward*. Elsevier B.V., 2023. doi: 10.1016/j.iot.2023.100721.

[24] X. Zhang *et al.*, "Risk prediction model of gestational diabetes mellitus based on nomogram in a Chinese population cohort study," *Sci. Rep.*, vol. 10, no. 1, pp. 1–7, 2020, doi: 10.1038/s41598-020-78164-x.

[25] G V Rakshitha and M. J. Anand, "IoT based Diabetes Detection using Machine Learning Algorithms," no. June, pp. 469–471, 2020.

[26] Y. Xiong *et al.*, "Prediction of gestational diabetes mellitus in the first 19 weeks of pregnancy using machine learning techniques," *J. Matern. Neonatal Med.*, vol. 35, no. 13, pp. 2457–2463, 2022, doi: 10.1080/14767058.2020.1786517.

[27] S. Amarnath, M. Selvamani, and V. Varadarajan, "Prognosis model for gestational diabetes using machine learning techniques," *Sensors Mater.*, vol. 33, no. 9, pp. 3011–3025, 2021, doi: 10.18494/SAM.2021.3119.

[28] S. Rani and M. Kumar, "Prediction of the mortality rate and framework for remote monitoring of pregnant women based on IoT," *Multimed. Tools Appl.*, vol. 80, no. 16, pp. 24555–24571, 2021, doi: 10.1007/s11042-021-10823-1.

[29] Z. Zhang, L. Yang, W. Han, Y. Wu, and L. Zhang, "Machine Learning Prediction Models for Gestational Diabetes Mellitus : Meta-analysis Corresponding Author :," vol. 24, doi: 10.2196/26634.

[30] S. Padhy, S. Dash, S. Routray, S. Ahmad, J. Nazeer, and A. Alam, "IoT-Based Hybrid Ensemble Machine Learning Model for Efficient Diabetes Mellitus Prediction," *Comput. Intell. Neurosci.*, vol. 2022, no. iii, 2022, doi: 10.1155/2022/2389636.

[31] N. El-Rashidy, N. E. ElSayed, A. El-Ghamry, and F. M. Talaat, "Prediction of gestational diabetes based on explainable deep learning and fog computing," *Soft Comput.*, vol. 26, no. 21, pp. 11435–11450, 2022, doi: 10.1007/s00500-022-07420-1.

[32] A. A. Nancy, D. Ravindran, P. M. D. Raj Vincent, K. Srinivasan, and D. Gutierrez Reina, "IoT-Cloud-Based Smart Healthcare Monitoring System for Heart Disease Prediction via Deep Learning," *Electronics*, vol. 11, no. 15, p. 2292, Jul. 2022, doi: 10.3390/electronics11152292.

[33] E. Yıldırım, M. Cicioğlu, and A. Çalhan, "Fog-cloud architecture-driven Internet of Medical Things framework for healthcare monitoring," *Med. Biol. Eng. Comput.*, no. 0123456789, 2023, doi: 10.1007/s11517-023-02776-4.

[34] A. Hennebelle, H. Materwala, and L. Ismail, "HealthEdge: A Machine Learning-Based Smart Healthcare

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023

Framework for Prediction of Type 2 Diabetes in an Integrated IoT, Edge, and Cloud Computing System," *14th Int. Conf. Ambient Syst. Networks Technol.*, vol. 00, 2023.

[35] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog Computing: A taxonomy, survey and future directions," *Internet of Things*, vol. 0, no. 9789811058608, pp. 103–130, 2018, doi: 10.1007/978-981-10-5861-5_5.

[36] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," *Big Data Cogn. Comput.*, vol. 2, no. 2, pp. 1–18, 2018, doi: 10.3390/bdcc2020010.

[37] M. Bhatia, "Fog Computing-inspired Smart Home Framework for Predictive Veterinary Healthcare," *Microprocess. Microsyst.*, vol. 78, p. 103227, 2020, doi: 10.1016/j.micpro.2020.103227.

[38] 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," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/1070697.

[39] J.-H. Hsieh, R.-C. Lee, K.-C. Hung, and M.-J. Shih, "Rapid and coding-efficient SPIHT algorithm for wavelet-based ECG data compression," *Integration*, vol. 60, pp. 248–256, Jan. 2018, doi: 10.1016/j.vlsi.2017.10.006.

[40] T. Y. Liu, K. J. Lin, and H. C. Wu, "ECG Data Encryption Then Compression Using Singular Value Decomposition," *IEEE J. Biomed. Heal. Informatics*, vol. 22, no. 3, pp. 707–713, May 2018, doi: 10.1109/JBHI.2017.2698498.

[41] 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, 2019, doi: 10.1016/j.jss.2019.04.050.

[42] S. Tuli *et al.*, "HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments," *Futur. Gener. Comput. Syst.*, vol. 104, pp. 187–200, Mar. 2020, doi: 10.1016/j.future.2019.10.043.

[43] A. Pati, M. Parhi, and B. K. Pattanayak, "HeartFog: Fog Computing Enabled Ensemble Deep Learning Framework for Automatic Heart Disease Diagnosis," 2022, pp. 39–53. doi: 10.1007/978-981-16-9873-6_4.

Doi: https://doi.org/10.54216/FPA.120206
Received: January 19, 2023 Revised: April 25, 2023 Accepted: June 19, 2023