HealthFaaS: AI-Based Smart Healthcare System for Heart Patients Using Serverless Computing

Muhammed Golec, Sukhpal Singh Gill, Ajith Kumar Parlikad, and Steve Uhlig

***Abstract*—Heart disease is one of the leading causes of death worldwide, and with early detection, mortality rates can be reduced. Well-known studies have shown that the latest artificial intelligence (AI) can be used to determine the risk of heart disease. However, existing studies did not consider dynamic scalability to get the best performance from these AI models in case of an increasing number of users. To solve this problem, we proposed an AI-powered smart healthcare framework called HealthFaaS, using the Internet of Things (IoT) and a Serverless Computing environment to reduce heart disease-related deaths and prevent financial losses by reducing misdiagnoses. HealthFaaS framework collects health data from users via IoT devices and sends it to AI models deployed on a Google Cloud Platform (GCP)-based serverless computing environment due to its advantages, such as dynamic scalability, less operational complexity, and a pay-as-you-go pricing model. The performance of five different AI models for heart disease risk detection is evaluated and compared based on key parameters, such as accuracy, precision, recall, *F*-Score, and AUC. Experimental results demonstrate that the light gradient boosting machine model gives the highest success in detecting heart diseases with an accuracy rate of 91.80%. Further, we have tested the performance of the HealthFaaS framework in terms of Quality-of-Service (QoS) parameters, such as throughput and latency against the increasing number of users and compared it with a non-serverless platform. In addition, we have also evaluated the cold start latency using a serverless platform which determined that the amount of memory and the software language makes a direct impact on the cold start latency.**

***Index Terms*—Artificial intelligence (AI), heart disease, Internet of Things (IoT), machine learning (ML), serverless computing, smart healthcare.**

I. Introduction

THE CHANGE in nutrition and physical activities brought about by the modern lifestyle has led to an increase in many diseases, the most important of which is heart disease. Heart disease is a general term that includes many diseases of cardiovascular origins, such as heart failure, heart defects, and heart attack [1]. Death from heart disease is the most significant cause of death in the U.K. and worldwide [2]. Recent attempts to reduce obesity and increase awareness of the causes behind heart disease have resulted in lower rates of heart disease, but the heart disease rate still remains high. Fig. 1 shows the top four causes of death worldwide as published by the World Health Organization (WHO) [3] which shows that ischemic heart disease ranks is on top with 8.9 million deaths.

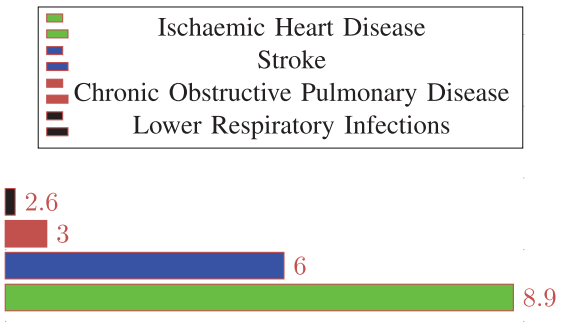

Fig. 1. Distribution of causes of death worldwide (millions) [3].

The most common complaint of patients with heart disease is chest pain. As a result of hematocrit, electrocardiography (ECG), and Thallium tests, a specialist makes a diagnosis as to whether the patient has heart disease [1]. Therefore, heart disease is difficult to detect with traditional methods (physical examination, etc.). Indeed, it acts like a silent killer, and patients are largely unaware they are being affected by it. For this reason, it is essential to diagnose heart diseases early and start treatment as soon as possible after detection. In light of recent developments in the Internet of Things (IoT) and artificial intelligence (AI) model techniques, studies that make early diagnoses are likely to reduce the number of fatal cases [4]. Therefore, there is a need to design new systems to detect heart diseases using IoT and sensors.

A. Motivation and Our Contributions

Advances in AI have led to several promising developments in smart healthcare systems. The cost of tests for the detection of heart disease, combined with the misdiagnose of heart disease, causes millions of pounds in financial loss in the health sector [5]. Patient data can be collected using IoT devices, and with the use of the latest machine learning

Manuscript received 25 August 2022; revised 16 December 2022 and 11 April 2023; accepted 15 May 2023. Date of publication 18 May 2023; date of current version 24 October 2023. The work of Muhammed Golec was supported by the Ministry of Education, Turkey. *(Corresponding author: Muhammed Golec.)*

Muhammed Golec is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, E1 4NS London, U.K., and also with the Department of Computer Engineering, Abdullah Gul University, 38080 Kayseri, Turkey (e-mail: m.golec@qmul.ac.uk).

Sukhpal Singh Gill and Steve Uhlig are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, E1 4NS London, U.K. (e-mail: s.s.gill@qmul.ac.uk; steve.uhlig@qmul.ac.uk).

Ajith Kumar Parlikad is with the Institute for Manufacturing, Department of Engineering, University of Cambridge, CB3 0FS Cambridge, U.K. (e-mail: aknp2@cam.ac.uk).

Digital Object Identifier 10.1109/JIOT.2023.3277500

(ML) techniques, the risk of heart disease can be successfully determined at an acceptable level [6]. Thus, using early and accurate diagnosis can reduce deaths due to heart disease and avoid unnecessary expenditures in health services. There is a need to determine the best ML model to predict heart disease early based on patients' health conditions. To achieve this, it is necessary to identify the ML model with the highest accuracy by comparing the performance of various latest ML models that have been successfully demonstrated in [7]. To achieve this, we have selected support vector machine (SVM), artificial neural network (ANN), extreme gradient boosting (XGBoost), gradient boosting machine (GBM), and light gradient boosting machine (LightGBM) models because another well-known research work [1] has shown that these models are most suitable for investigating the development of environmental-related cardiovascular disease and healthcare demand. In addition, the performance of these models was compared based on accuracy, precision, recall, F-Score, and AUC, and it has been identified that LightGBM outperformed with 91.8% accuracy.

Existing studies in disease detection with IoT need to consider dynamic scalability and respond to the user request with minimal latency and response time compared to nonserverless computing. Innovative systems that will provide high-computational power in terms of scalability are needed to meet multiple user demands at the same time, and process data from various IoT devices [7]. We deployed HealthFaaS using Google Cloud Platform (GCP)-Cloud Functions to build a serverless platform, and used Heroku to create a nonserverless platform and compared their performance. Performance evaluation shows that GCP-Cloud Functions-based serverless platform has a 477 p/sec throughput value to 500 concurrent requests and a response rate of 84 ms, a four times faster response rate and higher scalability than a nonserverless platform. Finally, the cold start latency of five different ML models has been evaluated for latency-sensitive IoT applications, such as patient follow-up. Experimental results show that the LightGBM model gives the least cold start latency with 1200 ms. Further, it has been determined that the amount of memory (RAM) and the software language makes a direct impact on the cold start latency.

The main contributions of this work are as follows.

1) To propose a new framework called HealthFaaS to reduce the number of fatal cases by early detection of heart diseases using ML/AI and IoT.
2) To identify the biomedical markers with the highest correlation that can be used to identify heart diseases for medical practitioners with feature selection methods.
3) To propose a model with advantages, such as dynamic scalability and a pay-as-you-go financial model to system users using a serverless platform.
4) To determine the most appropriate ML model for time-sensitive IoT applications by measuring the cold start delay caused by Serverless Computing.
5) To identify factors affecting cold start latency that should be considered when creating future IoT work environments using HealthFaaS.

The remainder of this article is structured as follows. Section II presents the studies on the diagnosis of heart disease using the data set used in our study. Section III describes the proposed methodology. Section IV presents the performance evaluations and experimental results. Section V concludes this article.

II. Related Work

With the recent development of AI, there is a great interest in research that detects diseases with ML models using IoT, fog, and cloud computing [14], [15]. In this section, we examine the studies that determine the risk of heart disease using ML models from the UCI Heart Disease data set [16] that we used in this work. In the first study proposed by Polat and Gunes, using ML, data sets are transformed from feature space to kernel space using linear and radial functions [8]. The F-score is calculated for the ML models. Further, they used the method called Kernel F-score feature selection (KFFS) in order not to negatively affect the accuracy rates of ML models. Accordingly, their accuracy rate in detecting the risk of heart disease is 83.70%. Spencer et al. proposed using the chi-squared feature evaluator to identify certain features in their work and use them to predict heart disease. In their study, the BayesNet algorithm and chi-squared feature evaluator combined achieved 85% accuracy in detecting heart disease [9]. However, different feature groups sometimes give conflicting results. In another study, Khourdifi and Bahaj [10] used quick feature selection to remove redundant variables from the data set before using ML models. By optimizing their models with the fast correlation-based feature selection (FCBF), particle swarm optimization (PSO), and ant colony optimization (ACO), they achieved the highest performance rates with the k-nearest neighbors (KNN) algorithm. In another study, optimal multi-nom logistic regression (OMLR) was used to determine the severity of the state of the heart [11]. This has yielded an accuracy of 92%. In some studies, combining different models has tried to increase the prediction accuracy for determining the risk of heart disease. One of these studies in the literature uses more than one method using Data Mining proposed by Tarawneh and Embarak [12]. In this work, an accuracy rate of 89.2% has been achieved. In one of the recent studies, it is a heart disease detection study using WEKA and knowledge extraction based on evolutionary learning (KEEL) open source tools. Another study [13] combines principal component analysis (PCA) and fuzzy logic, the accuracy rate was increased by decreasing the feature size in the data set. With the hybrid models used in the study, up to 94% accuracy rate has been achieved. Vilela et al. [17] proposed a fog computing-based study for real-time and latency-sensitive healthcare applications. Using IoT and some medical sensors, the authors tested energy consumption and delay metrics separately in cloud and fog environments. Since patients' health data are analyzed locally in fog computing, it provided advantages, such as data security, latency, and network usage compared to the cloud.

Table I shows the comparison of the proposed work (HealthFaaS) with existing works. As far as we know, no heart disease detection studies have been conducted which uses IoT

TABLE I
COMPARISON OF HEALTHFAAS WITH EXISTING WORKS

Study	Mechanism	Scalability	IoT	Serverless
[8]	KFFS	×	×	×
[9]	BayesNet	×	×	×
[10]	KNN	×	×	×
[11]	OMLR	×	×	×
[12]	Data Mining	×	×	×
[13]	GFS - LogisticBoost - C	×	×	×
HealthFaaS	LightGBM	✓	✓	✓

with Serverless Computing and AI in a single framework. None of the other studies focused on obtaining biomedical markers from users. This study can instantly receive data from users through IoT/wearable devices. In addition, previous research has focused only on the success of detecting heart disease with ML. We expanded this scope and deployed ML models on Serverless Computing. Thus, instant data received from users are sent to the ML model on the server. It is proposed to start the treatment process by sending information to health centers in case of disease detection using HealthFaaS. In this way, savings to the tune of millions of pounds in expenditures and heart disease-based deaths are expected to decrease.

III. METHODOLOGY

This section discusses the data set and defines the biomedical markers. Then, the general working mechanism of the proposed system will be examined. Finally, information about the serverless platform used in HealthFaaS will be given, and the performance evaluation metrics of the serverless platform will be explained.

A. Data Set and Biomedical Markers

The data set used in this work has been taken from the UCI ML Repository [16]. This data set contains 13 biomedical markers currently known to be important in the development of heart disease. These biomedical markers are as follows. *Age* represents the age of the patient. *Sex* represents the sex of the patient. 1 indicates that the patient is male, 0 indicates that the patient is female. *CP* represents whether the patient has chest pain. It indicates that there is chest pain for 1 and no chest pain for 0. *RestBP* indicates the resting blood pressure value of the patient. *Chol:* It represents the patient's cholesterol value. *FBS* represents the fasting blood sugar level of the person. It is denoted by 1 for values greater than 120 mg/dl, and 0 for smaller values. *rest ECG* represents the wave pulse for three different levels. *HeartBeat* represents the maximum heart rate. *Exang* represents exercise-induced angina. *OldPeak:* Depression includes exercise relative to rest. *Slope:* The condition of the person during the peak exercise segment. *CA:* The number of blood vessels colored by fluoroscopy. *Thal:* Four different values as results of Thallium tests. *Target:* It represents whether the patient has heart disease or not. It can only have two values: 1 represents heart disease and 0 represents non-heart disease situation.

B. HealthFaaS: System Architecture

Fig. 2 shows the HealthFaaS Framework. This study assumes that RestBP, Chol, FBS, rest ECG, HeartBeat,

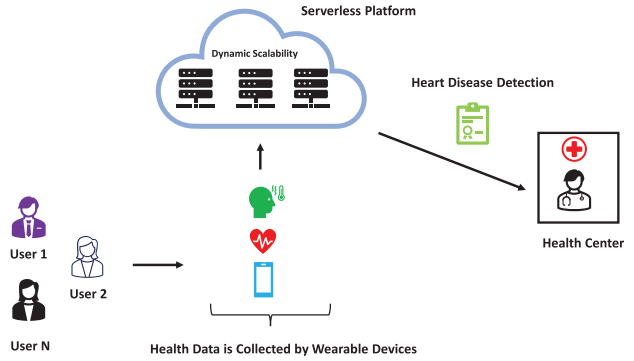

Fig. 2. HealthFaaS framework.

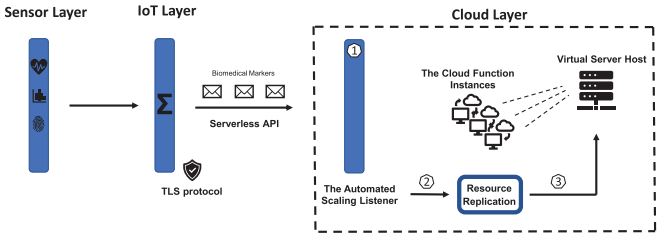

Fig. 3. HealthFaaS system architecture.

OldPeak, Slope, CA, and Thal biomedical markers are obtained from users via sensors in wearable devices. Other tokens are assumed to be entered manually. Raspberry Pi-4, an IoT device, was used to realize this scenario. Users' health data are sent to the serverless platform via an API and then sent to the previously deployed ML model. For the server side, GCP-based Cloud Functions which is a serverless platform is used [18]. The ML model deployed to the serverless platform has been trained with the previously given data set, and as this data set grows, the success rate can be increased by retraining the ML model in the cloud. Biomedical markers belonging to the user coming to the serverless platform if a heart disease is detected due to the estimation made in the ML model, information is sent to the nearest health center, and the user's treatment is started. In this way, it is planned to prevent deaths and financial losses due to heart disease by starting treatment with early diagnosis.

The system architecture of the HealthFaas Framework is explained in Fig. 3. HealthFaas consists of three different layers. In the Sensor Layer, biomedical markers are taken from the users and sent to the IoT Layer. Biomedical markers collected in IoT Layer are transmitted to Cloud Layer via API using secure TLS protocol [19]. The Automated Scaling Listener (1) follows the scalability policy predetermined by the cloud provider. When the number of requests from the IoT layer exceeds a certain threshold, it signals resource replication and starts the scalability process (2). The virtual server host divides the physical server into multiple servers using virtualization software. The primary purpose here is to use the server efficiently. Automated scaling listener continues to increase or decrease the resources in the cloud according to the demand from the IoT layer.

Algorithm 1 shows the pseudocode of the HealthFaaS Operating Mechanism. The biomedical markers (B_M) taken

Algorithm 1: HealthFaaS Operating Mechanism

Input: *Request*
Output: *Response*
Variables:
$B_M \leftarrow$ Biomedical Markers
$\Delta \leftarrow$ Prediction Result
$D_D \leftarrow$ Heart Disease
$N_D \leftarrow$ No Heart Disease
$R \leftarrow$ Request
$T_p \leftarrow$ Target Path
Begin
$\diamond$ *IoT Layer*
    R = Api.called()
    R.Path = T_p
    R = proxy.call(R)
    async{
        event = TransactionInfoEvent(
        Timestamp: time.Now()
        Biomedical Markers: $\sum B_M$}
$\diamond$ *Cloud Layer*
    if Δ == 1:
        Return D_D
    else:
        Return N_D
End

from the patients via the sensor are sent to the IoT layer. API is used to provide communication between IoT and Cloud Layer. B_M collected in the IoT Layer is sent to the cloud layer asynchronously with a request.B_M are given to the ML model that was previously deployed on a serverless platform, and it is expected to return a delta (Δ) as a result of prediction from the ML model. If Δ heart disease detection status equals the heart disease D_D, the patient's information is sent to the nearest health institute. If Δ is equal to the absence of heart disease N_D, no action is taken. Only the patient's health data is stored in the database to be followed by a specialist later. The HealthFaaS mechanism has no loops and only an if-else condition. Therefore, the time complexity is $O(1)$.

C. Serverless Computing Paradigm

Cloud computing technologies are becoming increasingly common to meet the need for high-processing power and extensive storage in IoT applications [7]. Cloud computing consists of three fundamental service models: 1) Infrastructure as a Service (IaaS); 2) Platform as a Service (PaaS); and 3) Function as a Service (FaaS) [20]. In the IaaS service model, a virtual server is created, and the cloud infrastructure and virtual server resources are allocated to the customer [21]. The customer has to deal with the operating system and runtime management. In the PaaS service model, since the cloud provider provides the system management, the customer only manages applications and data [22]. According to IaaS, infrastructure management is further abstracted. In Serverless Computing or FaaS (Function-as-a-Service), the customer is only responsible for the application's functionality [23], [24]. In other words, infrastructure management is more abstracted from the customer than the other two models. The differences between serverless computing and traditional cloud (IaaS and PaaS) can be summarized as follows.

1) Server management and infrastructure issues are entirely taken care of by the service provider: in serverless computing, customers are only concerned with the application's functionality, unlike IaaS and PaaS [25]. This way, they can spend most of their time developing code.
2) Serverless computing uses a pay-as-you-go financial model: customers only pay for the processing power and space they use. It means that, unlike IaaS and PaaS models, no fees are charged to customers during server idle periods [26].
3) With dynamic scalability, resources are automatically scaled if customers need them: unlike the IaaS and PaaS models, when using serverless computing, customers do not have to anticipate the storage and processing power that will be needed [23].

D. Serverless Computing Evaluation Metrics

The performance of serverless computing platform is evaluated using following three metrics.

1) *Throughput:* The average number of bits per second successfully delivered on the communication channel (bit/s) [24]. In communications, system designers often refer to throughput when evaluating the performance of a communications system.
2) *Average Response Rate (ARR):* It is the average of the time between the results of the requests sent from the client to the server and the time it takes to reach the client again [27]. It is a crucial metric for understanding the performance of the cloud service model used.
3) *The Cold Start Latency:* In serverless computing, the resources allocated for the execution of the function and the container are terminated upon the end of the function. In this way, no fee is paid for unused resources. This feature is called scaling to zero in serverless computing [28]. When the request comes again, a certain time is required for reassigning resources and creating containers which cause delays in applications. This delay is called cold start [29]. Cold start latency can be a problem for time-sensitive IoT applications, such as patient monitoring and autonomous vehicles [30].

IV. Performance Evaluation

This section will evaluate the performance of various ML models using metrics, such as accuracy, precision, recall, F-Score, and AUC to find out the model with the most successful prediction. Then we will find the effect values of the biomedical markers on our ML model and rank them in order of importance. In this way, we will determine which parameters healthcare professionals should pay more attention for heart disease detection studies. We will compare the throughput and ARR metrics to show the superiority of the serverless computing over nonserverless. Moreover, in the last section, we will compare the cold start latency of ML models

TABLE II
COMPARISON OF ML MODELS

Models	Accuracy	Precision	Recall	F-Score	Auc
GBM	83.60	89.65	78.78	83.87	94.04
ANN	85.24	85.29	87.87	86.56	81.01
XGBOOST	86.88	90.32	84.84	87.50	87.07
SVM	88.52	88.23	90.90	89.55	88.31
LightGBM	91.80	96.66	87.87	92.06	92.15

in serverless computing for time-sensitive IoT applications. Then we will observe the effect of cold start on Quality-of-Service (QoS) parameters. Finally, we will determine the factors affecting cold start while creating a working environment in serverless computing. In this way, we will try to reduce the cold start latency and its impact on in our application.

A. Machine Learning-Based Performance Analysis

By adjusting some hyperparameters of our ML algorithms that we used in our study, it has been ensured that they are operated in the most optimum way. These are as follows.

1) *GBM:* The booster = "gbtree," The learning_rate = 0.300000012, The max_depth = 6, and The n_estimators = 100.
2) *ANN:* The hidden_layer_sizes = 100, The alpha = 0.0001, and The activation = "relu."
3) *XGBoost:* The booster = "gbtree," The learning_rate = 0.300000012, The max_depth = 6, and The n_estimators = 100.
4) *SVM:* The kernels = "linear," The C hyperparameter = 1.0.
5) *LightGBM:* The boosting_type = "goss," The max_depth = −1, and The learning_rate = 0.1.

Table II shows the Accuracy, Precision, Recall, *F*-Score, and Auc values of five different ML models, and experimental results show that our model with the highest performance rate is LightGBM with 91.80% accuracy, and our model with the lowest performance rate is GBM with 83.60% accuracy. Since it will be sufficient to use only one ML model in the proposed study, LightGBM is the most appropriate model. Algorithms using Decision Trees can use one of two strategies as levelwise or leafwise. In the levelwise strategy, the balance of the tree is maintained while the tree grows. The leafwise strategy also continues from the leaves, reducing the division process loss. It differs from other boosting algorithms by using the LightGBM leafwise strategy. In this way, LightGBM has less error rate and learns faster.

B. Impact of Biomedical Markers on ML Models

Before the feature selection process, the correlation of the variables in the data set with the target variable was examined by creating a correlation matrix. Accordingly, "chol" and "fbs" variables were excluded from the data set because they had low correlation. After this process, the correlations of the variables with each other were examined and since the correlation value was "0," no additional variables were removed from the data set. "Mutual Information," "Pearson Correlation," and "Anova" models as feature selection methods were used in this study as they were successful in disease detection studies using ML in [31]. It was determined that the feature selection method with the highest accuracy rate for our data set was "Anova." After finding the order of importance of the variables using the "Anova" feature selection method, to find the feature set with the highest performance, 11, 9, and 7 features were selected, respectively, and ML models were established. Table III shows the accuracy and precision of ML models built based on these feature subsets. Accordingly, the highest accuracy and precision is achieved with the ML model established by creating the first nine features with the highest importance. In addition, these parameters used in heart disease detection studies can help to better understand the risk factors associated with heart disease.

TABLE III
CLASSIFICATIONS ON A DIFFERENT NUMBER OF FEATURES

Feature Number	Accuracy (%)	Precision (%)
11	81.96	82.90
9	83.60	83.71
7	77.37	76.75

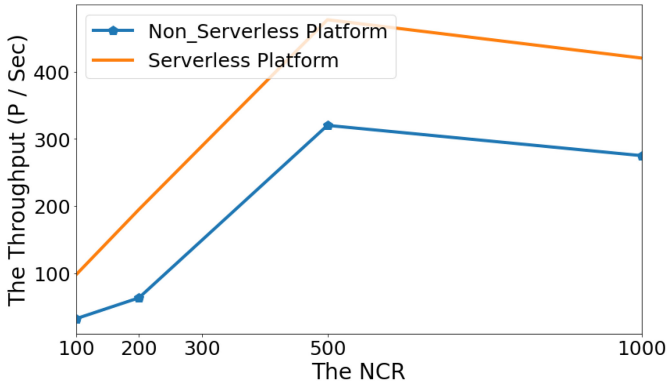

Fig. 4. Serverless and nonserverless comparison for throughput.

C. Performance Evaluation for Serverless Computing

Thanks to its dynamic scalability feature, serverless computing can respond to more users with higher throughput and a shorter ARR than nonserverless computing. To demonstrate this, we used GCP-Cloud Functions as a serverless computing platform and Heroku as a nonserverless platform. Then we deployed ML models on both platforms and tested their performance on both platforms. The workload is created via Apache JMeter using the UCI ML Repository data set in the HealthFaaS framework. The number of concurrent requests (NCRs) was sent at 100, 200, 300, 500, and 1000 per second to create the workload. Here, the NCRs represents the number of users using the system simultaneously. Accordingly, the data set variables are sent to the ML model on the server with the changing NCR. Moreover, the throughput and ARR given to this NCR are calculated. As the number of concurrent users increases, concurrent requests will also increase. Therefore, the throughput values also increase with the increase in the number of users accessing the platforms through an API. Fig. 4 shows the calculated throughput values against the NCR. Accordingly, the throughput obtained in 500 concurrent

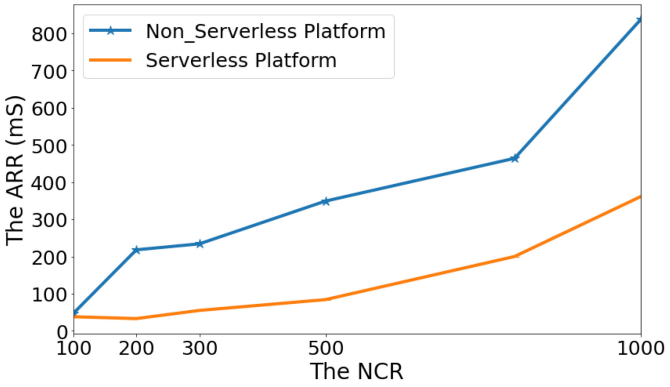

Fig. 5. Serverless and nonserverless comparison for ARR.

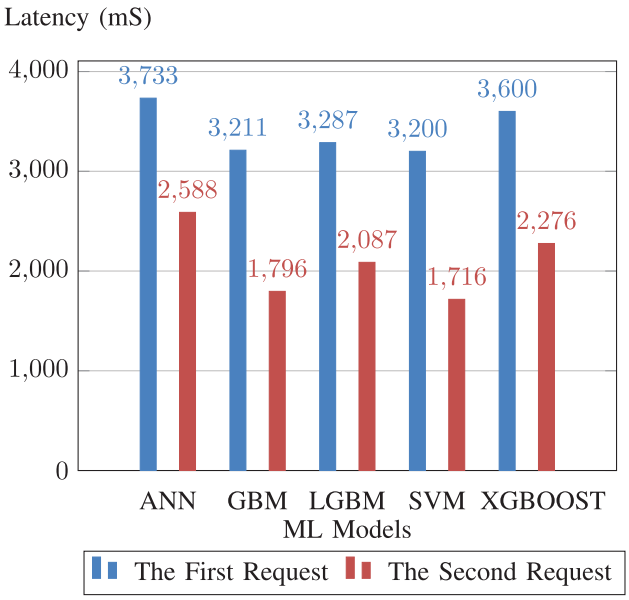

Fig. 6. Cold start latency for ML models.

requests reaches the maximum value for GCP-Cloud Functions and Heroku. Conflicts that arise when accessing the shared resource's disk storage, cache, and memory are called resource contention [32]. It was observed that the throughput value decreased after 500 NCR on both platforms due to the resource contention in the hardware resources. It has been noted that the throughput of GCP-Cloud Functions is much higher than the Heroku platform because GCP-Cloud Functions offers scalable services since it uses the FaaS service model of cloud computing. Fig. 5 shows the ARR values from the platforms against the NCR. As it can be seen, GCP-Cloud Functions can respond much faster than Heroku. Response time performance is a critical evaluation criterion for time-sensitive IoT applications. With the increase in the NCR in GCP-Cloud Functions, the response time is expected to increase. However, 100 NCR have a higher response rate than 200 NCR which is related to cold start latency.

In HealthFaaS, we measured the latency times on the client-side by sending two 100 NCR with J-Meter to determine the cold start latency values for five different ML models that we evaluated before. Cold start latency values are found by subtracting the response time calculated for the first request

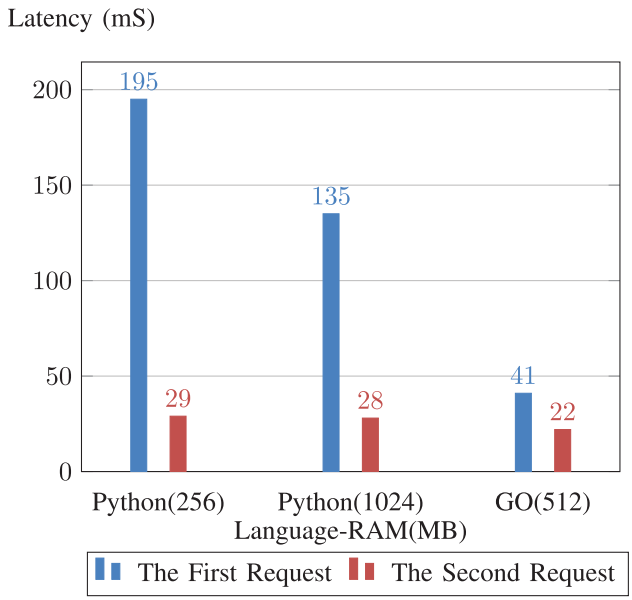

Fig. 7. Cold start latency by the factors.

from the response time calculated for the second request. Fig. 6 shows the cold start latency values calculated for five different ML models. Accordingly, the model with the highest cold start latency was SVM with 1538 ms, and the model with the least cold start latency was LGBM with 1124 ms. There are two types of scaling, horizontal and vertical scaling, to meet the increasing traffic demands [20]. In horizontal scaling, more devices are added to the infrastructure, and in vertical scaling, more processing power is added to a server. When Fig. 4 is examined carefully, it can be said that the resources are scaled horizontally in direct proportion to the increasing NCR. When the number of requests to the serverless platform exceeds a certain threshold, the resources will be scaled horizontally to meet the increasing traffic demands, and this will cause a new cold start latency like in Fig. 5. Although these latency durations are still a problem for time-sensitive scenarios, the academic community continues to work on cold start latency. In the last experiment, we identified the factors that affect cold start latency. In a serverless environment to be created by considering these factors, cold start latency can be decreased as much as possible. We created three different serverless computing environments using different ram amounts and software languages. Further, the same "hello world" function in these three environments were created and sent two consecutive requests. Fig. 7 shows the result of the experiment. Accordingly, factors such as the amount of RAM (main memory) allocated for the function and the software language used to affect the cold start latency time.

V. Conclusion and Future Work

With the promising developments in AI, we can see its applications in every aspect of our lives now. In recent years, the scientific community has resorted to AI to diagnose the disease. By accelerating the treatment process with early diagnosis, both the lives of patients can be saved and a significant reduction in expenditures for the health system can

be achieved. In this article, a new system called HealthFaaS has been proposed by combining the IoT and Serverless Computing technology that detects heart disease in patients using ML models and identified that LightGBM is the best model with 91.80% accuracy. With HealthFaaS, possible heart disease will be diagnosed in users as early as possible and the nearest health institution will be informed to save life. This would reduce the number of fatalities with the advantage of early diagnosis and a reduction in health expenditures. It is assumed that wearable devices are to be used to obtain health data from users. Thus, users' data can be followed instantly. Our work is deployed on the GCP-Cloud Functions due to its scalability feature. In this way, as the number of users using the system increases, the required resources and processing power can be easily provided in the cloud. In the future, HealthFaaS can be extended by incorporating the features related to security and user privacy to increase patients' confidence in the system. We have developed HealthFaaS as a benchmark through these AI models and future researchers can use other AI-based methodologies on edge for training and testing on prediction factors.

ACKNOWLEDGMENT

The authors gratefully acknowledge Google for their support of this research through their GCP research credit program.

REFERENCES

[1] H. Qiu, L. Luo, Z. Su, L. Zhou, L. Wang, and Y. Chen, "Machine learning approaches to predict peak demand days of cardiovascular admissions considering environmental exposure," *BMC Med. Inform. Decis. Making*, vol. 20, no. 1, p. 83, 2020.

[2] M. Roberts. "Unhealthy Britain: Nation's five big killers." Mar. 2013. [Online]. Available: https://www.bbc.com/news/health-21667065

[3] "The top 10 causes of death." Dec. 2020. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[4] S. S. Gill et al., "AI for next generation computing: Emerging trends and future directions," *Internet Things*, vol. 19, Aug. 2022, Art. no. 100514.

[5] S. Price. "Avoidable deficiencies in heart failure cost NHS £21m." Mar. 2020. [Online]. Available: https://www.healtheuropa.eu/avoidable-deficiencies-in-heart-failure-cost-nhs-21m/98696/

[6] A. Sitar-Tăut, D. Zdrenghea, D. Pop, and D. Sitar-Tăut, "Using machine learning algorithms in cardiovascular disease risk evaluation," *Age*, vol. 1, no. 4, p. 4, 2009.

[7] R. Singh and S. S. Gill, "Edge AI: A survey," *Internet Things Cyber-Phys. Syst.*, vol. 3, pp. 71–92, Feb. 2023.

[8] K. Polat and S. Güneş, "A new feature selection method on classification of medical datasets: Kernel F-score feature selection," *Expert Syst. Appl.*, vol. 36, no. 7, pp. 10367–10373, 2009.

[9] R. Spencer, F. Thabtah, N. Abdelhamid, and M. Thompson, "Exploring feature selection and classification methods for predicting heart disease," *Digit. Health*, vol. 6, Mar. 2020, Art. no. 205520762091477.

[10] Y. Khourdifi and M. Bahaj, "Heart disease prediction and classification using machine learning algorithms Optimized by particle swarm optimization and ant colony optimization," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 1, pp. 242–252, 2019.

[11] N. Satyanandam and C. Satyanarayana, "Heart disease detection using predictive optimization techniques," *Int. J. Image, Graph. Signal Process.*, vol. 11, no. 9, pp. 18–24, 2019.

[12] M. Tarawneh and O. Embarak, "Hybrid approach for heart disease prediction using data mining techniques," in *Advances in Internet, Data and Web Technologies*. Cham, Switzerland: Springer, 2019, pp. 447–454.

[13] F. Z. Abdeldjouad, M. Brahami, and N. Matta, "A hybrid approach for heart disease diagnosis and prediction using machine learning techniques," in *The Impact of Digital Technologies on Public Health in Developed and Developing Countries* (Lecture Notes in Computer Science 12157). Cham, Switzerland: Springer, 2020, pp. 299–306.

[14] H. Raj, M. Kumar, P. Kumar, A. Singh, and O. P. Verma, "Issues and challenges related to privacy and security in healthcare using IoT, fog, and cloud computing," in *Advanced Healthcare Systems: Empowering Physicians With IoT-Enabled Technologies*. Hoboken, NJ, USA: Wiley, 2022, pp. 21–32.

[15] S. Iftikhar et al., "AI-based fog and edge computing: A systematic review, taxonomy and future directions," *Internet Things*, vol. 21, Apr. 2023, Art. no. 100674.

[16] "Heart disease data set." 1988. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Heart+Disease

[17] P. H. Vilela, J. J. Rodrigues, P. Solic, K. Saleem, and V. Furtado, "Performance evaluation of a fog-assisted IoT solution for e-health applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 379–386, Aug. 2019.

[18] "Cloud functions   Google cloud." Accessed: Dec. 16, 2022. [Online]. Available: https://cloud.google.com/functions

[19] M. Golec, S. S. Gill, R. Bahsoon, and O. Rana, "BioSec: A biometric authentication framework for secure and private communication among edge devices in IoT and industry 4.0," *IEEE Consum. Electron. Mag.*, vol. 11, no. 2, pp. 51–56, Mar. 2022.

[20] I. Baldini et al., "Serverless computing: Current trends and open problems," in *Research Advances in Cloud Computing*. Singapore: Springer, 2017, pp. 1–20.

[21] H. K. Andi, "Analysis of serverless computing techniques in cloud software framework," *J. IoT Social Mobile Anal. Cloud*, vol. 3, no. 3, pp. 221–234, 2021.

[22] M. Wu, Z. Mi, and Y. Xia, "A survey on serverless computing and its implications for jointcloud computing," in *Proc. IEEE Int. Conf. Joint Cloud Comput.*, 2020, pp. 94–101.

[23] H. Shafiei, A. Khonsari, and P. Mousavi, "Serverless computing: A survey of opportunities, challenges and applications," 2019, *arXiv:1911.01296*.

[24] M. Golec, D. Chowdhury, S. Jaglan, S. S. Gill, and S. Uhlig, "AIBLOCK: Blockchain based lightweight framework for serverless computing using AI," in *Proc. 22nd IEEE Int. Symp. Cluster, Cloud Internet Comput. (CCGrid)*, 2022, pp. 886–892.

[25] P. Vahidinia, B. Farahani, and F. S. Aliee, "Cold start in serverless computing: Current trends and mitigation strategies," in *Proc. Int. Conf. Omni-Layer Intell. Syst.*, 2020, pp. 1–7.

[26] L. F. Albuquerque Jr., F. S. Ferraz, R. Oliveira, and S. Galdino, "Function-as-a-service x platform-as-a-service: Towards a comparative study on FaaS and PaaS," in *Proc. ICSEA*, 2017, pp. 206–212.

[27] Z. Tong, X. Deng, J. Mei, B. Liu, and K. Li, "Response time and energy consumption co-offloading with SLRTA algorithm in cloud–edge collaborative computing," *Future Gener. Comput. Syst.*, vol. 129, pp. 64–76, Apr. 2022.

[28] P. Castro, V. Ishakian, V. Muthusamy, and A. Slominski, "The rise of serverless computing," *Commun. ACM*, vol. 62, no. 12, pp. 44–54, 2019.

[29] P. Silva, D. Fireman, and T. E. Pereira, "Prebaking functions to warm the serverless cold start," in *Proc. 21st Int. Middleware Conf.*, 2020, pp. 1–13.

[30] A. Bhattacharjee, A. D. Chhokra, Z. Kang, H. Sun, A. Gokhale, and G. Karsai, "BARISTA: Efficient and scalable serverless serving system for deep learning prediction services," in *Proc. IEEE Int. Conf. Cloud Eng. (IC2E)*, 2019, pp. 23–33.

[31] M. Golec, R. Ozturac, Z. Pooranian, S. S. Gill, and R. Buyya, "iFaaSBus: A security and privacy based lightweight framework for serverless computing using IoT and machine learning," *IEEE Trans. Ind. Informat.*, vol. 18, no. 5, pp. 3522–3529, May 2022.

[32] Z. Li et al., "Amoeba: QoS-awareness and reduced resource usage of microservices with serverless computing," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. (IPDPS)*, 2020, pp. 399–408.

Muhammed Golec received the M.Sc. degree (Distinction) in computer science from Queen Mary University of London, London, U.K., in 2020, through the Ministry of Education Scholarship, where he is currently pursuing the Ph.D. degree in computer science.

He has published articles in prominent journals and conferences, such as IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, *Internet of Things Journal* (Elsevier), *IEEE Consumer Electronics Magazine*, and IEEE CCGRID. His research interests include cloud computing, serverless computing, AI, and security and privacy.

Sukhpal Singh Gill received the Doctor of Philosophy (Ph.D.) degree in computer science from Thapar University, Patiala, India, in 2016.

He is a Lecturer (Assistant Professor) of Cloud Computing with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. Prior to his present stint, he has held positions as a Research Associate with Lancaster University, Lancaster, U.K., and also as a Postdoctoral Research Fellow with CLOUDS Laboratory, The University of Melbourne, Parkville, VIC, Australia. He has published in prominent international journals and conferences, such as IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE INTERNET OF THINGS JOURNAL, *Journal of Systems and Software/Future Generation Computer Systems* (Elsevier), IEEE/ACM UCC, and IEEE CCGRID. His research interests include cloud computing, fog computing, IoT, and energy efficiency. For further information, please visit: http://www.ssgill.me.

Dr. Gill is serving as an Associate Editor for IEEE INTERNET OF THINGS JOURNAL, *Internet of Things Journal* (Elsevier), *Transactions on Emerging Telecommunications Technologies* (Wiley), and *IET Networks*.

Ajith Kumar Parlikad received the Ph.D. degree from the University of Cambridge, Cambridge, U.K., in August 2006.

He is a Professor of Asset Management with the Engineering Department, University of Cambridge. He is with the Institute for Manufacturing, Cambridge, where he is the Head of the Asset Management Research Group. He is a Fellow and Tutor with Hughes Hall, University of Cambridge. He leads research activities on engineering asset management and maintenance. His particular focus is examining how asset information can be used to improve asset performance through effective decision making. His research interests are digital twins, engineering asset management, Internet of Things, Industry 4.0, cyber–physical systems, reliability and maintenance engineering, and value of information.

Steve Uhlig received the Ph.D. degree in applied sciences from the University of Louvain, Ottignies-Louvain-la-Neuve, Belgium, in 2004.

Prior to joining Queen Mary University of London, London, U.K., he was a Senior Research Scientist with Deutsche Telekom Laboratories, Technische Universität Berlin, Berlin, Germany. Since January 2012, he has been the Professor of Networks and the Head of the Networks Research Group, Queen Mary University of London. From 2012 to 2016, he was a Guest Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His current research interests include Internet measurements, software-defined networking, and content delivery.