



# IOT Based Healthcare Monitoring System for Diabetes Prediction using Extreme Gradient Boosting Techniques

3790

**T.Ramyaveni M.C.A, M.Phil,**

Research Scholar, Department of Computer Science, A.V.V.M Sri Pushpam College (Autonomous) Poondi, Thanjavur. Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.

**Dr.V.Maniraj**

Associate Professor & Research Supervisor, Department of Computer Science, A.V.V.M Sri Pushpam College (Autonomous) Poondi, Thanjavur. Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.

## Abstract

Human health issues must be closely examined and addressed with proper medications. Chronic illnesses such as diabetes, heart disease (HD), cancer, and chronic respiratory disease are the main causes of death worldwide. The previous 10 years have seen a lot of research into healthcare services and their technology advancements. To be more specific, the Internet of Things (IoT) has showed promise in connecting a variety of medical devices, sensors, and healthcare specialists in order to provide high-quality medical treatment in a remote place. Patient safety has improved, healthcare expenses have decreased, healthcare services have become more accessible, and the healthcare industry's operational efficiency has increased. In this paper, a diabetic patient monitoring strategy is proposed that uses an IoT-based machine learning technique called eXtreme Gradient Boosting (XGB) to support in diabetes diagnosis and classification. A successful implementation of any classifier requires proper hyperparameter optimization. This work employed Bayesian optimization, which is a very effective method for hyper-parameter optimization, to optimize the hyper-parameters of XGBoost. The efficacy of the suggested method is assessed in terms of accuracy, specificity, sensitivity, and F1score. It outperforms better than the other existing algorithms.

**Keywords:** Heart disease, Intrnet of Things, eXtreme Gradient Boosting (XGB)

**Number: 10.14704/nq.2022.20.7.NQ33469**

**Neuro Quantology 2022; 20(7):3790-3798**

## 1. Introduction

Diabetes mellitus is a metabolic, never-ending illness caused by elevated blood sugar levels in the circulatory system. It has an effect on several regions of the body, including pancreas malfunction, heart disease risk, hypertension, renal failures, pancreatic concerns, nerve damage, foot issues, ketoacidosis, visual disturbing impacts, and other eye problems, such as waterfalls and glaucoma [1]. In 2017, around 451 million persons worldwide were diagnosed with diabetes. It is estimated that by 2045, there will be about 693 million diabetic individuals worldwide, with half of the population remaining undiagnosed [2]. Global diabetes treatment expenses were projected at 760 billion dollars in 2019 and are projected to climb to 845 billion dollars by 2045 [3]. It will become extremely difficult for medical experts to deliver good

treatment manually due to the rapid growth in diabetic cases and growing complexity in enormous data sets of diabetic patients. As a result, medical data mining can be successfully utilized for better diabetic patient diagnoses because it allows diabetes to be detected at an earlier stage [4]. Early detection of the disease lowers medical expenditures and lowers the chance of individuals developing more serious health problems [5]. Early detection of risk-prone people is recommended in medical guidelines, as is patients' proactive self-monitoring of their lifestyle to reduce risk factors [6]. In the therapeutic sector, classification approaches are often used to organize data into distinct groupings, with some requiring practically an individual classifier. Many analysts are doing tests for diabetes diagnosis using various grouping calculations of machine learning



algorithms such as J48, Decision Tree, Naive Bayes, Support Vector Machine (SVM), Decision Tree, Ada Boosting, and so on [7]. The provision of treatment depends on an accurate and timely detection of diseases [8]. Dealing with missing data, which can occur due to sensor issues, human error, or during transmission between system pieces in different places, such as cloud servers, is one of the most difficult issues that data analytics faces [9]. Furthermore, missing data causes bias, which leads to erroneous results [10]. From a technical standpoint, the Internet of Things (IoT) is fast gaining traction in a variety of fields, particularly personalized healthcare [11]. Integration of IoT sensor networks with new technologies provides efficient solutions for dealing with the dynamic and complicated nature of sensor data. Machine learning and deep learning algorithms are also intriguing options for analyzing IoT sensor data. Incorporating these data analysis approaches yields deep insights into sensor data as well as valuable knowledge about hidden data patterns and subsequent decision-making [12].

## 2. Literature survey

Khan et al. [13] offered a comprehensive evaluation of the state-of-the-art in data mining for diabetes diagnosis and prediction. They investigate and explore data mining-based diagnosis and prediction methods in the field of diabetes glycemic management. Kayal Vizhi and Aman Dash [14] employed a variety of machine learning methods, including linear regression, xgboost, decision tree, support vector machine, and random forest, to determine the overall efficiency and accuracy in predicting whether or not a human will develop diabetes. Deberneh Henock M and Intaek Kim [15] constructed a machine learning (ML) model to predict T2D recurrence in the following year ( $Y + 1$ ). Key features were chosen for the prediction model using ANOVA tests, chi-squared tests, and recursive feature reduction approaches. On the basis of these characteristics, logistic regression, random forest, support vector machine, XGBoost, and ensemble machine learning techniques were used to predict whether the outcome will be normal (non-diabetic), prediabetes, or diabetes. Naz Huma and Sachin Ahuja [16] provided a methodology for diabetes prediction using Naive Bayes (NB), Artificial Neural Network (ANN), Decision Tree (DT), and Deep Learning (DL) are functional classifiers that attain accuracy in the

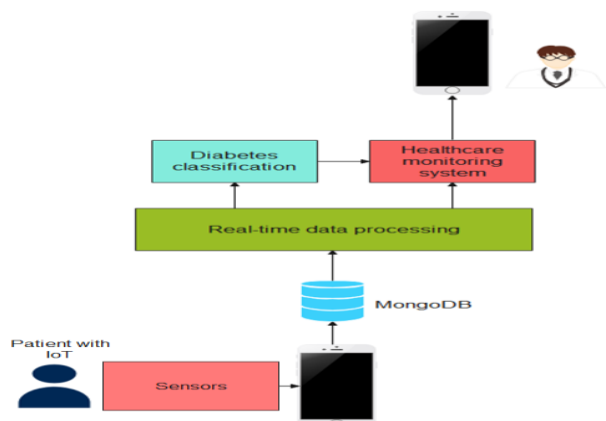
90–98% range. Muhammad et al [17] used random forest, support vector machine, K-nearest neighbour, logistic regression, naive Bayes, and gradient booting techniques to create predictive supervised machine learning models. T2DM was developed by Liying et al. [18], who used six machine learning algorithms: classification and regression tree (CART), support vector machine (SVM), logistic regression (LR), random forest (RF), artificial neural networks (ANN), and gradient boosting machine (GBM). The GBM model outperformed the others (AUC: 0.872 with laboratory data and 0.817 without laboratory data). Badiuzzaman et al [19] used multiple Machine Learning approaches to develop a model with a few constraints based on the PIMA dataset to analyse diabetes patients as well as diabetes detection. In this study, they compared decision trees, K-nearest neighbour, random forest, and Nave Bayes, and the results reveal that random forest and Nave Bayes are the best classifiers. Machine learning approaches with feature selection were given by Oladosu et al [20], which can diagnose diabetic disease at an early stage. The result demonstrates that feature selection aids in the development of a better model by preventing overfitting and removing unnecessary data. Adua et al [21] used four machine learning (ML) classification techniques to predict Type 2 Diabetes Mellitus: K-Nearest Neighbor (KNN), Support Vector Machines (SVM), Nave-Bayes (NB), and Decision Tree (DT) (T2DM). According to Sohail et al. [22], a decision-making classifier (J48) is applied over a data-mining platform (Weka) to measure accuracy and linear regression on classification results to forecast cost/benefit ratio in diabetes mellitus patients along with prevalence using a decision-making classifier (J48). Choubey et al. [23] offered an indigenously developed and effective diabetes diagnostic system. The proposed methodology is divided into two stages: The first method uses Logistic Regression, ID3 DT, K-Nearest Neighbor, C4.5 DT, and Naive Bayes to classify data. The second strategy uses PCA and PSO algorithms for feature reduction. Anuradha et al. [24] developed a unique fuzzy rule miner (ANT FDCSM) for diabetic patient diagnosis obtained from an ant colony meta-heuristic. To increase the performance of the standard ant colony optimization induced decision tree classifier, a few key changes have been proposed. Using stacked auto encoders, Kannadasan et al [25]



established a Deep Neural Network framework for diabetes data classification. Stack auto encoders are used to extract features from the dataset, and the dataset is classified using the softmax layer. In addition, the network is fine-tuned using supervised backpropagation with the training dataset.

### 3. Methodology

Diabetes has become a big problem for people all around the world. The method of predicting diabetes is incredibly difficult. It can only be done successfully if the doctor is well-versed in the disease and has extensive expertise with it. The effectiveness of the remote health monitoring system for elderly patients who require long-term care has increased significantly as a result of the advancement of the Internet of Things and its medical applications. To help patients better self-manage their chronic disease, a customized healthcare monitoring system was developed. The system's core idea is to collect data from users using IoT sensors and then send it over a wireless network to a remote service platform. Personal data about users (gender, height, age, and other details) is also collected. An Android app was created to gather data from the sensors as well as personal input from the user. Sensor and personal details are wirelessly transported to a secure remote server, where real-time data processing is implemented, allowing the system to manage massive amounts of sensor data. The NoSQL MongoDB Atlas was used as the database to hold the sensor data from the patients in this study.



**Figure 1** Architecture of the healthcare monitoring system for diabetic patients

The patient's diabetes is predicted and classified using a machine learning technique. The analysis' findings are subsequently communicated to the medical team via a web-based healthcare

monitoring system, and the patient's final personal healthcare therapy is administered. Apache Kafka is used for real-time data processing since it is a streaming platform that can handle streams of events and provides low-latency, high-throughput, fault-tolerant publication, and subscription pipelines. The architecture of the healthcare monitoring system is depicted in Figure 1. The patient's data is formatted in JSON and wirelessly delivered to the secure remote server using the created Android app. The sensor data is received using a Node.js web application. **3792**

#### I. Patients with IoT

The proposed work's first phase begins with the sensor device being attached to the patient's body. The sensor on the IoT device will sense the values of the patient's body and transfer them to the Health Care Monitoring (HCM) application. The patient's sensed data is stored in a Mongo database, which is then processed for diabetes prediction and classification.

#### II. Authentication

This is a crucial step in granting authorized people access. At the start of the application, the authentication procedure is always active. Different systems may require different types of qualifications to verify a user's identification. The credential is typically in the form of a password that is kept private and is only known by the individual and the system.

#### III. Registration

The patient enters their information into the HCM application (hospital app) or website during this phase. The server then combines the login and password of the enrolled patient with the help of a SC. An SC is a cryptographic technique in which plain text units are swapped with ciphertext according to a predetermined system. Single letters, two letters, three letters, a blend of the preceding, and so on could be used as units. The text is deciphered by the receiver using the inverse substitution.

#### IV. Login

Logging in is the process of gaining access to an application, which is usually located on a remote computer. Every system login phase necessitates the user's username and password (registered). When users log into the system, their authentication is confirmed by performing a verification.



## V. Verification

The system checks to see if the user is a registered one. The administrator of the HCM application checks the users by examining their data. The entered username is combined with the password at this phase, and the SC is applied.

## VI. Upload Patient's Data

The user must first register their details, such as login and password, before submitting the material. The patient will use his or her unique username and password to log onto the hospital website or app after registering their information. This patient information is saved on the cloud server (CS) as well as the hospital database. After a patient has been registered, they can upload their information. For the purpose of prediction, the patients' data is uploaded to the application. For data processing, the data is stored in a cloud database. The system goes through training and testing in order to accomplish classification. Preprocessing and feature selection are carried out during training, and classification is performed. The sensor information from the CS is examined after training and classed as normal or abnormal.

### 3.1 Diabetes Prediction

Early disease detection allows individuals to be treated before their condition worsens. This research used data from the PIMA Indians Diabetes Database at the National Institute of Diabetes and Digestive and Kidney Diseases [31]. All of the patients in this dataset are Pima Indian women who are at least 21 years old. The dataset's goal is to diagnose whether a patient has diabetes using diagnostic metrics included in the collection. These files contain the records of 768 patients, 500 of whom tested normal and 268 of whom tested positive. The goal of using the data was to diagnose whether or not a patient had developed diabetes. The dataset contains eight attributes, such as the number of pregnancies, diastolic blood pressure, triceps skinfold, 2 h serum insulin, 2 h glucose tolerance, body mass index, diabetes pedigree function, 2 h serum insulin, and age.

### 3.2 Pre-processing

The initial step in the process of classification is pre-processing. Data preprocessing is a necessary step for cleaning data and preparing it for a machine learning model, which improves the model's accuracy and efficiency. It also has an

effect on the model's ability to generalize. It entails replacing missing values, eliminating redundancy, and eliminating null values. In the redundancy removal procedure, the data size is reduced by removing redundant and null values from the dataset.

### 3.3 Feature Selection

Understanding how the model's features contribute to prediction is critical to optimizing the model's performance. When utilizing the XGBoost model for diabetes prediction, pay attention to the most significant features. Feature selection becomes more important in data sets with a large number of variables and features.

---

#### Algorithm 1 for feature selection using XGBoost

---

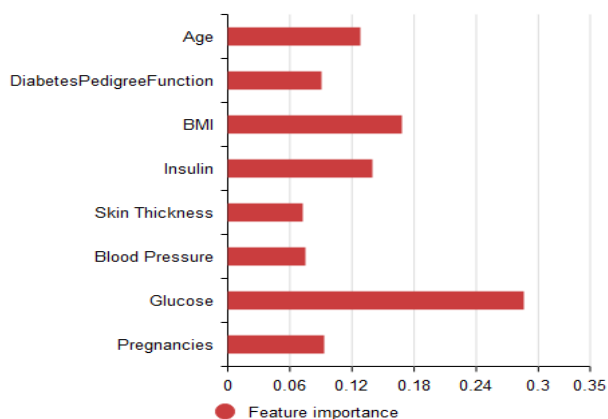
**Input:**  $F_{Preprocessed}$  ( $F_n = f_1, f_2, \dots, f_n$ )

**Output:**  $I_f$ , the selected feature vector

- Step 1: Load the pre-processed feature vector
- Step 2: Create an empty set S to save the score of feature
- Step 3: Initiate a GradientBoostingClassifier as GBC
- Step 4: fit GBC
- Step 5: Generate  $FS_n$
- Step 6: Determine the threshold T
- Step 7:
  - for n from  $F_{Preprocessed}$  do
  - if  $FS_n \geq T$ ) then
  - append  $FS_n$  into S
  - end if
- end for
- Step 5: Using scores in set S, generate  $I_f$

---

Understanding the datasets and selecting the attributes that will yield the key data required to infer the knowledge sought is part of the feature selection process. It will eliminate insignificant variables and increase classification accuracy and performance. It aids in the simplification of the model by reducing the number of parameters, the reduction of training time, the reduction of overfilling by improving generalization, and the avoidance of the dimensionality constraint.

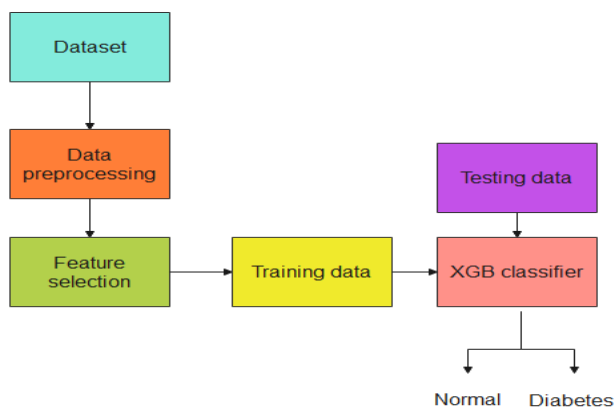


**Figure 2** Feature importance score of features.

Glucose, BMI, Insulin, and Age are the features with the highest relevance score. The classification model is trained using these features. A classification technique (also known as a classifier) is a method for creating classification models from a set of data. The dataset for diabetes prediction is trained using machine learning techniques, and analysis is performed based on the training. The eXtreme Gradient Boosting classifier was employed in this strategy.

### 3.4 Classification using XGB

XGBoost is a unique machine learning algorithm that has attracted a lot of interest due to its superior learning effect and quick training speed. The XGBoost technique is a step forward from the gradient boosting decision tree (GBDT) and can be used to solve classification and regression problems. XGBoost is a boosting tree algorithm that combines numerous weak classifiers to generate a powerful classifier. The classification and regression tree (CART) model is the tree model it employs.



**Figure 3** Classification of diabetes disease.

Only the first derivative is used in the GBDT algorithm. The current nth tree's value is proportional to the residual of the first n 1 trees, which is challenging to do. XGBoost uses the power of the loss function's second-order Taylor expansion and adds a regularization term to balance the model's complexity and the loss function's decrease. It looks for the best overall solution and, to some extent, avoids overfitting. Using a multi-thread CPU, XGBoost can automatically construct gradient tree boosting techniques in parallel, making the methods run faster and improving algorithm precision. Using the group of decision tree, the prediction can be done and it is given as,

$$\hat{y}_i = \sum_{d=1}^d f_d(x_i), f_d \in F \tag{1}$$

where,  $\hat{y}_i$  represent the prediction of the i-th instance at the d-th tree,  $x_i$  represent the i-th instance sample of the training dataset. The value of the d-th tree is  $f_d(x_i)$  and all the values of the decision trees are represented by the function F. To train the model, loss function should be optimized. The loss function can be given as,

$$L = \sum_{i=1}^n L(\hat{y}_i, y_i) \tag{2}$$

Because XGBoost is a decision tree-based technique, it uses a number of tree-related hyper-parameters, such as subsample and max depth, to avoid overfitting and increase model performance. In addition, the learning rate controls the model's tree weighting and is used to slow down the model's rate of adaption to the training data. The regularization notion in the XGBoost objective function benefits in the selection of prediction functions and the control of model complexity. The objective function of the XGBoost is obtained by combining the loss function with the regularization term. The loss function controls the model's predictive power, whereas the regularization term controls the model's simplicity. We can define objective function of the XGBoost as shown in equation (3)

$$Obj = \sum_{i=1}^n L(\hat{y}_i, y_i) + \sum_{i=1}^d P(f_i) \tag{3}$$

where L represents the loss function which determines the compatibility of the model on training data; predicted label is denoted by  $\hat{y}_i$  and  $y_i$  denotes the actual label. P (f) is responsible for penalizing the complexity of the functions of the training tree. It also handles the overfitting problem. To define the complexity, first, we need to define the function of tree  $f(x)$ . Thus, the objective function is expressed as



$$f(x) = S_{m(x)}, S \in P^N \tag{4}$$

Here  $S$  represent the leaves scores vector,  $m$  represent a mapping function which maps data instances to the corresponding leaf, and the number of leaves is represented by  $N$ .

$$P(f) = \gamma T + \alpha (||w||) + \frac{1}{2} \lambda (||w||^2) \tag{5}$$

where  $\gamma$  and  $\lambda$  are the hyper-parameters or constant coefficients, each leaf value is represented by  $\gamma$ , and the total number of leaves in the tree is represented by  $T$ .  $||w||^2$  denotes the L2-norm of the weight of the leaf controls by  $k$  term and  $||w||$  denotes the L1- norm of the weight of the leaf controls by a term. L2 regularization (controlled by the `reg_lambda` term) encourages the weights to be small, whereas L1 regularization (controlled by the `reg_alpha` term) encourages sparsity. The minimum loss reduction is determined by the hyper-parameter  $c$  (gamma) for further partition. The objective function of XGBoost is optimised using gradient descent. Our model is an additive model, which means that it adds a tree to the model every time the forecast result equals the sum of the existing and new trees. So, at the  $t$ -th step, among these equations, the objective at each step is calculated by Eq. (6)

$$Obj^{(t)} = \sum_{i=1}^n L(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + P(f_t) + Const \tag{6}$$

Because it is impossible to calculate all combinations of trees at once, the tree structure is built by calculating the leaf scores, regularization, and objective function at each level. This tree structure will be reused in following iterations, reducing computing complexity dramatically. In addition, during the node splitting process, the gain of each feature is determined. It recursively determines the optimal dividing point till it reaches the maximum depth. The nodes are then pruned out in a bottom-up fashion with a negative gain. This is how XGBoost classifies data and passes deep into trees.

### 3.5 Hyper-parameter optimization

In machine learning, hyper-parameter optimization is the process of determining which

hyper-parameters for a given machine learning algorithm yield the best results when tested on a validation set. Manual search, random search, grid search, and Bayesian optimization are four typical methods for hyper parameter optimization. With a huge number of hyper parameters, manual search is impossible. For hyper parameter finding, Bayesian optimization is more comprehensive and time-efficient. It is not necessary to change all of the hyper parameters. Learning rate, `reg_lambda`, `min child weight`, `cosample bytree`, `max depth`, `subsample`, `n estimators`, `gamma`, and `reg_alpha` are the nine parameters we tuned in this paper. The learning rate improves the model's stability and robustness, while the `min child weight`, `max depth`, `subsample`, `colsample bytree`, and `gamma` reduce over-fitting.

### 4. Result and discussion

We use evaluation measures including accuracy, sensitivity, specificity, and F1-score to assess the efficacy of the suggested strategy. The percentage of correctly identified subjects is known as accuracy. The ratio of those who have the condition and test positive is known as sensitivity. The fraction of those who do not have the disease who test negative is known as specificity. The terms recall and sensitivity are interchangeable. The fraction of participants accurately recognized as positive out of the total number of subjects identified as positive is known as precision. A harmonic mean of precision and recall is the F1-Score.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

$$Specificity = \frac{TN}{TN+FP}$$

$$Sensitivity = Recall = \frac{TP}{TP+FN}$$

$$F1 \text{ score} = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

where TP and FP denotes the number of correctly and wrongly classified subject having diabetes, respectively. Similarly, TN and FN denotes the number of correctly and wrongly classified subject not having diabetes, respectively.



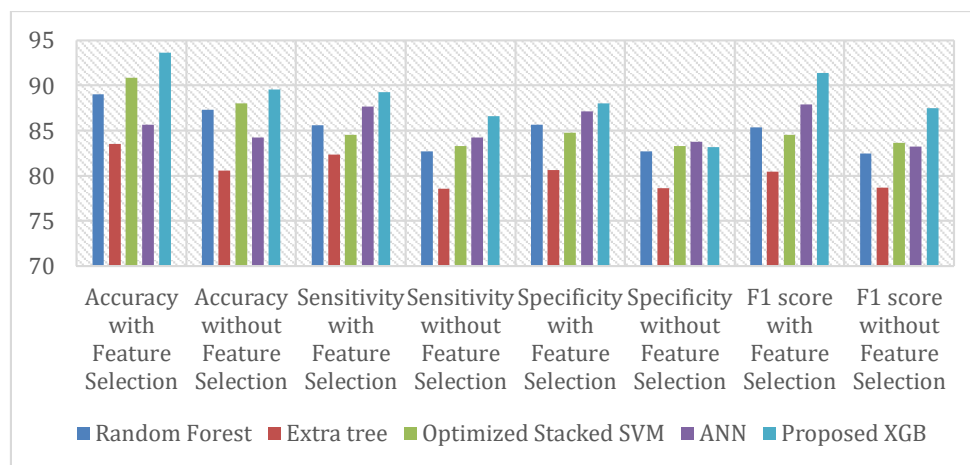
**Table 3** Simulation results of XGBoost Hyper-parameter optimization using Bayesian optimization on the dataset.

Iteration	learning_rate	max_depth	gamma	reg_lambda	min_child_weight	colsample_bytree	n_estimators	subsample	reg_alpha
0	0.484528	7	1.874507	42.63936	0	0.962845	939	0.693546	16.53864
1	0.159467	3	0.029756	18.54924	3	1	1000	0.835219	23.03671
2	0.5	5	1.853981	0	1	0.855632	838	0.837193	43.76553
3	0.398527	10	0	73.02846	2	1	921	1	0
4	0.369284	6	4.884592	100	5	0.659023	825	0.917342	62.85408
5	0.5	8	2.639748	1.53716	5	0.745291	811	0.834261	100
6	0.072641	5	0	44.26491	2	1	912	0.845621	0
7	0.247004	10	2.693265	62.74834	0	0.756209	842	1	44.57566
8	0.5	9	0.710375	100	1	0.684536	956	0.668733	73.91856
9	0.5	4	0	33.03578	3	0.538560	837	1	100

The hyper-parameters of the XGBoost algorithm were improved using the Bayesian Optimization algorithm, and then performance analysis was performed utilizing those optimized hyper-parameters. We find the best set of parameters using Bayesian Optimization algorithm iterations. Because no significant progress is expected, we configured the model to halt after ten iterations.

**Comparative analysis with other Machine learning models**

To demonstrate the efficacy of the proposed model, the experimental results are compared to RF, Optimized Stacked SVM, ANN, and Extra Tree classifiers. The proposed algorithm has the highest accuracy of 93.63% among other machine learning algorithms which are shown in Figure 4.



**Figure 4** Performance of the proposed model with existing techniques

It is clear from the above figure 4, that the proposed method performance is better compared with the other existing techniques in terms of accuracy, f1 score, specificity and sensitivity.

**5. Conclusion**

This work proposed a health monitoring system for diabetes diagnosis using IoT and machine learning technique. The proposed method for the

disease prediction used XGBoost for feature selection and classification. The proposed method used Bayesian optimization as a hyper-parameter optimization technique which is proved to be a very efficient technique to get the best hyper-parameters. The proposed method is evaluated using different evaluation metrics, namely sensitivity, specificity, F1-score, accuracy. We observed that our tree-based ensemble proposed method performs better than the other three



previously proposed methods. The proposed model outperforms other models by 93.63% of accuracy, 89.26% sensitivity, 88.03% specificity and 91.39% f1 score. Based on experimental results, we can conclude that the suggested diagnostic method would improve the quality of decision-making during the diagnosis of the diabetes.

## References

1. Sneha N., and Tarun Gangil. "Analysis of diabetes mellitus for early prediction using optimal features selection." *Journal of Big Data* 6, no. 1 (2019). doi:10.1186/s40537-019-0175-6.
2. T. Mahboob Alam et al., "A model for early prediction of diabetes," *Inform. Med. Unlocked*, vol. 16, no. 100204, p. 100204, 2019.
3. Chaves, Luís, and Gonçalo Marques. "Data Mining Techniques for Early Diagnosis of Diabetes: A Comparative Study." *Applied Sciences* 11, no. 5 (2021), 2218. doi:10.3390/app11052218.
4. Mishra, S., Tripathy, H. K., Mallick, P. K., Bhoi, A. K., & Barsocchi, P. (2020). EAGA-MLP—An enhanced and adaptive hybrid classification model for diabetes diagnosis. *Sensors*, 20(14), 4036. <https://doi.org/10.3390/s20144036>
5. Lai, Hang, Huaxiong Huang, Karim Keshavjee, Aziz Guergachi, and Xin Gao. "Predictive models for diabetes mellitus using machine learning techniques." *BMC Endocrine Disorders* 19, no. 1 (2019). doi:10.1186/s12902-019-0436-6.
6. Ramesh Jayroop, Raafat Aburukba, and Assim Sagahyoon. "A remote healthcare monitoring framework for diabetes prediction using machine learning." *Healthcare Technology Letters*, 2021. doi:10.1049/htl2.12010.
7. Mahabub, Atik. "A robust voting approach for diabetes prediction using traditional machine learning techniques." *SN Applied Sciences* 1, no. 12 (2019). doi:10.1007/s42452-019-1759-7.
8. Li, X., Zhang, J., & Safara, F. (2021). Improving the accuracy of diabetes diagnosis applications through a hybrid feature selection algorithm. *Neural Processing Letters*. <https://doi.org/10.1007/s11063-021-10491-0>
9. Torkey, H., Ibrahim, E., Hemdan, E. E., El-Sayed, A., & Shouman, M. A. (2021). Diabetes classification application with efficient missing and outliers data handling algorithms. *Complex & Intelligent Systems*. <https://doi.org/10.1007/s40747-021-00349-2>
10. Dzulkalnine, M. F., & Sallehuddin, R. (2019). Missing data imputation with fuzzy feature selection for diabetes dataset. *SN Applied Sciences*, 1(4). <https://doi.org/10.1007/s42452-019-0383-x>
11. Wan, J., A. A. H. Al-awlaqi, M., Li, M., O'Grady, M., Gu, X., Wang, J., & Cao, N. (2018). Wearable IoT enabled real-time health monitoring system. *EURASIP Journal on Wireless Communications and Networking*, 2018(1). <https://doi.org/10.1186/s13638-018-1308-x>
12. Krishnamurthi, R., Kumar, A., Gopinathan, D., Nayyar, A., & Qureshi, B. (2020). An overview of IoT sensor data processing, fusion, and analysis techniques. *Sensors*, 20(21), 6076. <https://doi.org/10.3390/s20216076>
13. Khan Farrukh A., Khan Zeb, Mabrook Al-Rakhami, Abdelouahid Derhab, and Syed A. Bukhari. "Detection and Prediction of Diabetes Using Data Mining: A Comprehensive Review." *IEEE Access* 9 (2021), 43711-43735. doi:10.1109/access.2021.3059343.
14. Dr. Kayal Vizhi, and Aman Dash. (2020). Diabetes Prediction Using Machine Learning. *International Journal of Advanced Science and Technology*, 29(06), 2842 - 2852. doi: <http://sersc.org/journals/index.php/IJAST/article/view/13795>
15. Deberneh Henock M., and Intaek Kim. "Prediction of Type 2 Diabetes Based on Machine Learning Algorithm." *International Journal of Environmental Research and Public Health* 18, no. 6 (2021), 3317. doi:10.3390/ijerph18063317.
16. Naz Huma, and Sachin Ahuja. "Deep learning approach for diabetes prediction using PIMA Indian dataset." *Journal of Diabetes &*



- Metabolic Disorders 19, no. 1 (2020), 391-403. doi:10.1007/s40200-020-00520-5.
17. Muhammad, L. J., Ebrahim A. Algehyne, and Sani S. Usman. "Predictive Supervised Machine Learning Models for Diabetes Mellitus." SN Computer Science 1, no. 5 (2020). doi:10.1007/s42979-020-00250-8.
18. Liying Zhang, Yikang Wang, Miaomiao Niu, Chongjian Wang, and Zhenfei Wang. "Machine learning for characterizing risk of type 2 diabetes mellitus in a rural Chinese population: the Henan Rural Cohort Study." Scientific Reports 10, no. 1 (2020). doi:10.1038/s41598-020-61123-x.
19. Badiuzzaman Pranto, Sk. M. Mehnaz, Esha B. Mahid, Imran M. Sadman, Ahsanur Rahman, and Sifat Momen. "Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh." Information 11, no. 8 (2020), 374. doi:10.3390/info11080374.
20. Oladosu Oladimeji, Abimbola Oladimeji, and Olayanju Oladimeji. "Classification models for likelihood prediction of diabetes at early stage using feature selection." Applied Computing and Informatics ahead-of-print, no. ahead-of-print (2021). <https://doi:10.1108/aci-01-2021-0022>.
21. Adua, E., Kolog, E. A., Afrifa-Yamoah, E., Amankwah, B., Obirikorang, C., Anto, E. O., Acheampong, E., Wang, W., & Tetteh, A. Y. (2021). Predictive model and feature importance for early detection of type II diabetes mellitus. <https://doi.org/10.21203/rs.3.rs-150169/v1>
22. Sohail, M. N., Jiadong, R., Uba, M. M., Irshad, M., Iqbal, W., Arshad, J., & John, A. V. (2019). A hybrid forecast cost benefit classification of diabetes mellitus prevalence based on epidemiological study on real-life patient's data. Scientific Reports, 9(1). <https://doi.org/10.1038/s41598-019-46631-9>
23. Choubey, D. K., Kumar, P., Tripathi, S., & Kumar, S. (2019). Performance evaluation of classification methods with PCA and PSO for diabetes. Network Modeling Analysis in Health Informatics and Bioinformatics, 9(1). <https://doi.org/10.1007/s13721-019-0210-8>
24. Anuradha, Singh, A., & Gupta, G. (2019). ANT\_FDCCSM: A novel fuzzy rule miner derived from ant colony meta-heuristic for diagnosis of diabetic patients. Journal of Intelligent & Fuzzy Systems, 36(1), 747-760. <https://doi.org/10.3233/jifs-172240>
25. Kannadasan, K., Edla, D. R., & Kuppili, V. (2019). Type 2 diabetes data classification using stacked autoencoders in deep neural networks. Clinical Epidemiology and Global Health, 7(4), 530-535. <https://doi.org/10.1016/j.cegh.2018.12.004>
26. Pima Indians diabetes database. (n.d.). Kaggle: Your Machine Learning and Data Science Community. <https://www.kaggle.com/uciml/pima-indians-diabetes-database>

