IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 9 December 2024, accepted 29 December 2024, date of publication 10 January 2025, date of current version 21 January 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3528033

RESEARCH ARTICLE

A Cloud-Based Optimized Ensemble Model for Risk Prediction of Diabetic Progression–An Azure Machine Learning Perspective

V. K. DALIYA AND T. K. RAMESH, (Member, IEEE)
Department of Electronics and Communication Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Bengaluru 560035, India
Corresponding author: T. K. Ramesh (tk_ramesh@blr.amrita.edu)

ABSTRACT The application of Machine Learning for predictive analysis in healthcare, particularly for diseases like diabetes, has proven highly beneficial. This study introduces an optimized Light Gradient-Boosting Machine (Light GBM) and K-Nearest Neighbour (KNN) based ensemble algorithm for predicting diabetic progression of Type 2 Diabetes, classifying it as high or low risk, using patient health parameters and serum measurements. Our model uses LightGBM, a rapid and efficient gradient boosting framework, coupled with KNN, which uses proximity to classify data points. The proposed model uses various optimization techniques, such as 10 fold cross validation, grid search method etc. to get the best results out of the ensemble model. As the model combines optimized version of LightGBM and KNN through a voting classifier which uses soft voting technique to find the final class, it utilizes the predictive capabilities of both the methods in an effective manner. The experiment is performed and implemented in Microsoft's Azure cloud, using Azure Machine Learning service, that leverages the advantages of cloud computing with respect to scalability, security and its potential integration possibilities into IoT-based smart healthcare systems.This aspect highlights its versatility and impact with respect to remote monitoring of patients as well. The ensemble achieves an 83.2% Area Under the Curve (AUC) of Receiver Operating Characteristics (ROC) score, indicating good classification efficiency. It produced 75% accuracy as well. The proposed model is compared with other classification and ensemble models, showcasing its superiority against other models.The ensemble is also tested with some meta heuristic optimization methods, which produced comparable scores. The method's effectiveness is validated against another risk prediction dataset, proving its reliability. The model's accurate predictions can aid individuals in understanding disease progression risks and guide medical professionals in intervention strategies.

INDEX TERMS Diabetic prediction, ensemble learning, KNN, LightGBM, Machine learning, voting classifier, azure cloud, azure machine learning.

I. INTRODUCTION

Machine Learning (ML) is pivotal in various sectors linked by Internet-connected devices. IoT applications harness ML to manage vast data streams from connected devices, benefiting industries like transportation, healthcare, and E-commerce [1]. Health IoT systems play major role in fast diagnosis of diseases and can help with immediate medical help, using connected devices which can monitor health parameters of patients [2], [3]. ML in healthcare IoT enables remote patient monitoring and emergency response [4]. ML algorithms, a core data mining tool, learn from pre-processed data for predictive analysis or clustering [5]. They employ mathematical techniques like probability and optimization. ML algorithms cover regression and classification [5], with binary classification adopted here. There are different algorithms based on gradient boosting method such as XGBoost, AdaBoost, CatBoost etc. in literature [6].

The associate editor coordinating the review of this manuscript and approving it for publication was R. K. Tripathy.

Gradient boosting, including LightGBM, gained attention in the recent years. In our work, we have performed Diabetic progression risk prediction using optimized LightGBM and KNN based technique.

Diabetes is a disease which prevails all over the world. It is a concern of the entire world to keep the progression of the disease under control. Diabetes prediction, considers various factors such as a person's health parameters, blood serum measurements, exercise, hereditary factors, and other diseases, which affect the diabetic status [7]. Diabetes Mellitus is classified into Type 1 Diabetes (T1D) and Type 2 Diabetes (T2D). While T1D is caused due to lack of insulin which helps in regulating blood glucose levels, T2D is a result of inability of insulin to regulate blood glucose properly [8]. Generally T1D is associated with genetic factors and it is usually diagnosed in childhood or adolescent stage. It requires life long insulin therapy. T2D is mostly caused due to issues related to lifestyle as well as genetic reasons. T2D is the most common type of Diabetes detected in adults. In this study, we are discussing the prediction of Diabetic Progression risk of T2D, using the optimized ML model.

While traditional optimization in Machine Learning focuses on adjusting the parameter of a chosen model, choosing the right model and further adjusting the parameters to obtain optimum results becomes the overall optimization strategy. This study proposes a LightGBM and KNN based ensemble for healthcare, predicting diabetic progression. Here, the data is tested with various individual ML algorithms initially. After analyzing the capabilities of such algorithms, efficiency and robustness of LightGBM algorithm in classification has been identified. Similarly, the use of ensemble methods in ML by combining models with different skills have been analyzed. Here, combination of LightGBM and memory based KNN has been proven to be producing the best results. The ensemble's optimized versions is given to a voting classifier, which predicts the outcome using soft voting method, yielding high accuracy. Optimization of data starts from using different methods in preprocessing and utilizing the best methods to make it ready for further processing using ML techniques. Data Scaling, feature analysis, cross validation, hyper parameter tuning, comparison with other models and validation with other datasets are the methods employed in the study as part of the overall optimization.

Similarly, if the best fitting model obtained can be implemented as a web service in cloud, it becomes easy for Health IoT systems to utilize the service. Here, we have experimented and implemented the models in Microsoft's Azure cloud. Implementing Machine Learning methods on the Azure cloud offers several significant advantages. Firstly, Azure provides a scalable and flexible infrastructure, allowing users to easily scale up or down based on their computational needs. This scalability ensures that Machine Learning models can handle large datasets and complex computational efficiently. Azure offers a range of ML services in Azure Machine Learning platform, which streamline the development and deployment of ML models [9]. Azure ensures security of the data used in machine learning projects, which protects the data privacy. It helps in integration with other tools and services of Azure as well. ML analysis here, aids assessment and control, especially with cloud-based model deployment. Implementation in the Azure Cloud as web service helps in optimizing the resource usage and it enables the service to be a part of real time prediction framework of Health IoT system. Hence, usage of Azure cloud services for experimentation and implementation further adds to the optimization technique, in terms of resource allocation, scalability etc.

Literature explores diabetes prediction and progression of the disease, using various ML models such as Neural Networks (NN), Support Vector Machines (SVM), Ensemble approach, Genetic Algorithms etc. [10]. Section A elaborates on related work explored.

A. RELATED WORK

Papers on progression of diabetes and related diseases are explored from literature. Similarly immediate prediction of diabetes and classification of patients, if they are diabetic or not, based on vital parameters were also analysed. The main findings are listed here. Study [11], developed a ML model, which predicts occurrence of Type-2 diabetes(T2D), in the next year, using variables in the current year. Dataset is collected from a private medical institute as Electronic Health Record (EHR) from the year 2013 to 2018 (535169 instances). 2 years of continuous annual medical check up during follow up period was also analyzed. Resultant features identified were Fasting Plasma Glucose, HbA1C, Triglycerides, BMI, Gamma GTP, Age, Uric acid, sex, smoking, drinking, physical activity and family history. Algorithm used are Logistic Regression (LR), Random Forest (RF), SVM, XGBoost and Ensemble. Using these, output is categorized into normal, pre-diabetic and diabetic classes. Forecasting of T2D was reasonably good. Reference [12] discusses about usage of Artificial Intelligence (AI) to support clinical judgement in medicine. New predictive model for Diabetic Kidney Disease (DKD), using AI, processing natural language and other data with Big data based on Electronic Medical Record (EMR) of 64,059 patients was proposed. AI extracted raw features from previous 6 months. 24 factors were considered and 6 months' DKD aggravation using convolution autoencoder was measured. AI could predict DKD aggravation with 71% accuracy. In [13], an analysis of identification in prior; of those at high risk of progression from pre-diabetes to diabetes, was carried out, utilizing patient data from EMR. Here ML model was developed using a Gradient Boosted Tree model. Data from The Health Improvement Network database (THIN) was considered with external validation of Canadian Apple Tree and Israeli Maccabi Health Service data set. Prediction capability was compared with that of LR method. The ML model was proven superior over LR. The contributing signals identified were Age, gender, glucose, HbA1c, BMI, triglycerides, ALT, WBC, HDL and Aspirin usage. The article [14], used a total of 206 patients with risk factors to study for a diagnostic of new onset of T2D. Risk factors considered were Essential hypertension, Obesity and if the candidate is first degree relative with a diagnosis of diabetes. Glucometry was performed using continuous glucose monitoring system. Detrended Fluctuation Analysis (DFA) was emerged as predictor for the development of diabetes. In a multivariate analysis, Fasting glucose, HbA1c and DFA emerged as significant factors. The work [15], deals with DKD in relation with diabetes. Ability to predict the progression of DKD using classically described risk markers is poor. Established risk factors in T1D and T2D for this disease are HbA1c, systolic BP, Albinurin grade, Duration of diabetes, Microvascular Amplication, Retinopathy, Family history, Age and serum uric acid. In article [16], novel ML algorithm was experimented on 5 years EHR data to get insights on disease progression of T2D, with AUC of 76%. The risk factors identified for the same are Blood Glucose(BG), Blood Pressure, Triglycerides, Lipid disorders and socio economic factors. In follow up, 15% of patients are detected with T2D and 46% transitioned to pre-diabetes. In [17]; as relatives of type 1 diabetes (T1D) are at enhanced risk of developing diabetes, investigation was carried out on the mode of onset of hyperglycemia and how insulin sensitivity and beta cell function contribute to the progression of the disease. Oral Glucose Tolerance Test (OGTT) was used. In high-risk relatives, beta cell glucose sensitivity is impaired and is a strong predictor of diabetic progression. From these progression based papers, the factors which affect the diabetic progression were identified. Some of the papers exclusively discussed effect of diabetic progression to diseases such as kidney diseases.

Discussion about immediate prediction and classification of diabetes using ML models are listed in the following papers. In article [18], a method was introduced using weighted Support Vector Regression (SVR) with the Differential Evolution (DE) algorithm to predict BG levels in T1D patients. The DE algorithm optimizes SVR parameters, claiming advantages in accuracy, adaptiveness, robustness, and practicality over other techniques. Paper [19], proposed a deep learning approach for diabetic prediction, focusing on Continuous Glucose Monitoring (CGM) data for children under 18, using time series data for prediction after 30 minutes. Results demonstrated superior performance over shallow learning methods. In [20], four ML techniques, based on Feed forward Neural Network (FNN), Self-Organizing Map (SOM), Neuro-fuzzy network with wavelet as activation function (WFNN) and Linear Regression Method (LRM), were applied to glucose prediction, with the Self-Organizing Map (SOM) showing excellence in hypoglycemic and hyperglycemic ranges, while all models performed well in euglycaemic range. The study [21] used feature ranking with RF and RReliefs combined with SVR and Genetic Programming (GP) models for T1D prediction. RF and RReliefs resulted in equally predictive feature ranking. In [22], SVM achieved 78% accuracy on a Pima Indian diabetic dataset, with sensitivity of 80% and specificity of 76.5%. Study [23] explored multiple algorithms, with Improved Diabetes Prediction Algorithm (IDPA) achieving 96.07% accuracy, surpassing other models, such as K-means with LR(94.6%), Hierarchical clustering(HC) with LR(94.1%), HC with SVM (94.1%), HC with Decision Tree (90%). T1D and T2D prediction using time series data was discussed in [24], within the context of Internet of Medical Things (IOMT). It devised precise person-specific short-term prediction models, using array of features. Data of 40 T1D patients were analysed using Random Forest technique and it produced glucose levels within a 30-minute prediction horizon with an average error of 18.60 mg/dL for six-hour data, and 26.21 mg/dL for a 45-minute prediction horizon. Data from 10 Type 2 DM patients were also taken into consideration for validation. It demonstrated the potential of IoMT-based methodologies for continuous monitoring and management of Diabetes.

From the progression based papers it was found that factors such as Age, gender, glucose, triglycerides, WBC, HDL, HbA1C, BMI, Thyroid Stimulating Hormmone etc. are major factors influencing the progression of T2D in patients. Lifestyle habits such as smoking, drinking lack of physical activity etc. can also influence the disease. Genetic factors are also found to play a role in T2D. The papers which discussed immediate prediction of diabetes, used BG information, diet and insulin as input parameters while only a few considered effects of physical activity and other body conditions. It is also found that there is rise in error rates with increase in the prediction horizon. Some of the work discussed specific cases for the diseases such as DKD which can occur due to prolonged diabetes. Detailed analysis and optimization is a requirement in progression analysis. Moreover experimentation and implementation in cloud to leverage its advantages is missing in the literature. Considering the insights obtained in terms of features influencing progression in T2D and the research gaps identified from the literature review, we have investigated diabetic disease progression after one year in the work. Here, 10 input features are considered and dataset of 442 patients are analyzed. The features considered include Age, Sex, BMI, Average Blood Pressure and 6 blood serum measurements which influence the disease progression. The major contributions of the research is discussed in section B.

B. MAJOR CONTRIBUTIONS

- The paper analyzes the long term progression of diabetes, which helps patients to get proper medical advice on their future aspects of the disease and prevent the risk.
- The use of Ensemble approach contributes to obtain the best outcome compared to the individual methods.
- The AUC of ROC is obtained as 83.2%, which indicates promising classification efficiency.
- The method considers 10 intrinsic features that influence the disease, which connotes good authenticity.

- Comparison with other classification methods and optimization techniques, demonstrates high degree of efficiency and accuracy of the proposed technique.
- The use of one of the best gradient boosting method called LightGBM in combination with memory based KNN in the optimized way, adds up the advantages of both.
- Validation of the model with another dataset, which produced promising results, implies the robustness of the model.
- The cloud implementation promotes the chances of making the prediction available as part of the Health IoT system. The method can be extended for the prediction of any other disease as well.
- Implementing machine learning methods on Azure cloud offers scalability, ease of development, security, and integration capabilities, making it an ideal platform to leverage machine learning solutions for real time scenarios.
- Comparison of Cloud implementation with Edge computing and Hybrid architecture emphasizes the pros and cons of each method in Health IoT implementation.

The rest of the paper is organized as follows. Section II discusses theoretical aspects of basic algorithms used. Section III elaborates on the dataset used. System architecture of the proposed model is narrated in section IV. Section V describes proposed method. Result discussion is presented in section VI. Conclusion and future work is discussed in section VII, followed by references.

II. BASIC ALGORITHMS USED

As the basic algorithms used in this paper are LightGBM and KNN; theoretical aspects of these methods are discussed in part A and B of this section. Section C briefs about Voting Classifier, which predicts the final outcome.

A. LIGHT GRADIENT-BOOSTING MACHINE

Gradient boosting is a robust predictive modeling technique. Introduced to enhance weak learners, it employs decision trees as the base [25]. Adaboost, the initial version, assigns higher weights to data which are tough to classify and lower weights to data which are easy to classify, adapting until an effective approach is achieved. Gradient boosting employs a loss function, weak learner, and an additive model to minimize the loss. Any differentiable loss function can be used here. Weak learners, like decision trees, are iteratively added; using gradient descent to minimize errors. LightGBM is highly efficient and fast. It uses less memory and provides good accuracy. It has large data handling capabilities as well. Unlike other methods, LightGBM uses leaf-wise splitting [26] (shown in Fig. 1), enhancing loss reduction. The growth of the tree takes place based on the expandable leaf nodes. In Fig. 1, red nodes expand whereas blue ones don't.

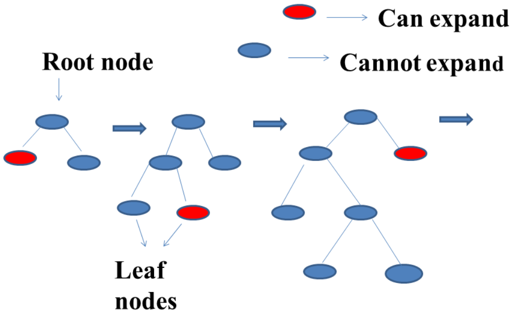


FIGURE 1. Leafwise tree growth.

Leaf-wise split in LightGBM may increase its complexity and may result in overfitting. It can be restricted by specifying the maximum depth by which the tree grows. Number of leaves set must be lesser than $2^{Maximumdepth}$.

B. K-NEAREST NEIGHBOUR

KNN, a proximity-based ML technique, assigns data points to classes based on their feature similarity. It calculates Euclidean distance between test and training data points, arranging them by distance [27]. By choosing 'K' neighbors, a majority-class voting determines the data's predicted class. If the number of classes in the problem is even number, it is wise to choose odd number as K value, as it will break the tie which might occur [27]. The Euclidean distance formula is provided (equation 1).

$$D(w, w') = \sqrt{([(w_1 - w'_1)^2 + \ldots + (w_n - w'_n)]^2])} \tag{1}$$

where $D(w, w')$ represents the Euclidean distance and $w_1, w_2, w_3 w_4, \ldots w_n$ represents the values of each input data in the test set and $w'_1, w'_2, w'_3 \ldots wn'$ signify the corresponding data values in the training set. Optimal K balances neighbour count and decision boundary [28]. The paper uses Grid search method, which scans through all combinations of parameters and stores the values. It helps determine the optimum K value for better accuracy. For binary classification, probabilities for each class are calculated (equation 2) based on count of data points.

$$P(class = x) = \frac{(N(class = x))}{(N(class = x) + N(class = y))} \tag{2}$$

Here, $P(class = x)$ represents the probability of the data point belonging to class x and $N(class = x/y)$ represents the number of data points in the class x or y. Here, x and y represent classes '0' and '1' respectively.

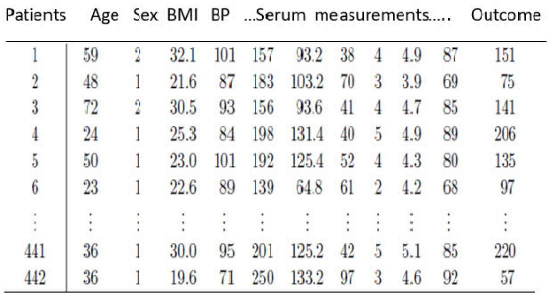
C. VOTING CLASSIFIER

Voting classifier finds the final class of a combined model either by taking majority voting (hard voting) or by finding the average of the class probabilities of the individual models (soft voting) [29]. The study used a voting classifier with soft voting method, giving equal weights for both the models. Description of the dataset used is discussed in section III.

III. DATASET DESCRIPTION

The proposed model uses a dataset which consists of 442 patients' health parameters such as Age, Sex, BMI, Average Blood Pressure (BP) and 6 blood serum measurements: Low density Lipoproteins (LDL), High Density Lipoproteins (HDL), Total Cholesterol (TC), Thyroid Stimulating hormone (TG), Blood Glucose level (Glu) and serum concentration of Lamotrigine (LTG) which is a diabetic neuropathy medicine consumed by the patients. These values are fed as input parameters to the model, which cover most of the intrinsic factors that influence diabetes in a person. The outcome of the dataset is a set of values which signifies the Quantitative Measurement of the Diabetic Disease Progression (QMDDP) obtained; one year after measuring the input parameters.

The sample of the dataset is depicted in Fig. 2 [30]. The Serum Measurements in Fig. 2 indicate the 6 blood serum measurements explained and Outcome represents QMDDP.



Patients	Age	Sex	BMI	BP	...Serum measurements.....						Outcome
1	59	2	32.1	101	157	93.2	38	4	4.9	87	151
2	48	1	21.6	87	183	103.2	70	3	3.9	69	75
3	72	2	30.5	93	156	93.6	41	4	4.7	85	141
4	24	1	25.3	84	198	131.4	40	5	4.9	89	206
5	50	1	23.0	101	192	125.4	52	4	4.3	80	135
6	23	1	22.6	89	139	64.8	61	2	4.2	68	97
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
441	36	1	30.0	95	201	125.2	42	5	5.1	85	220
442	36	1	19.6	71	250	133.2	97	3	4.6	92	57

FIGURE 2. **Sample of the original dataset used.**

The dataset is publically available at the link, https://www4.stat.ncsu.edu/ boos/var.select/diabetes.html. The dataset was first used in the paper, "Least Angle Regression" [30]. In this paper, Least Angle Regression method was analyzed mathematically. The paper also discussed about how their method can be used as a base learner in boosting method.

According to Fig. 2, as the outcome of the dataset consists of quantitative values, it is required to convert the same to '0' and '1', since we are considering binary classification here. '0' represents low risk of diabetic progression (class 'low') and '1' represents high risk of diabetic progression (class 'high'). To convert the quantitative values of the outcome into high and low classes, value '140' is taken as the threshold, as it is a good predictor of diabetic risk, according to findings from various studies such as articles [31], [32], and [33]

The analysis in study [31] established a Random Blood Glucose (RBG) cut-off of 140.5 mgdl (7.8 mmol/L), which is linked to an HbA1c level of 6.5% (48 mmol/mol). The authors concluded that an RBG level of $\geq$ 140 mg/dl (7.8 mmol/L) can be used for identifying individuals who may require a confirmatory Oral Glucose Tolerance Test for diagnosis of diabetes. RBG measurements can serve as an alternative when HbA1c or timed glucose tests are impractical for large-scale screening of undiagnosed diabetes. In research [32], a criterion of 2-hour plasma glucose $\geq$ 200 mg/dl (11.1 mmol/L) indicated that a Random Capillary Blood Glucose cut-off of 140 mg/dl (7.7 mmol/L) provided the best sensitivity and specificity. According to article [33], impaired glucose tolerance testing is prescribed for postprandial glucose levels between 140 – 199 mg/dL at 2 hours. Based on these findings, the threshold value of '140' has been selected for categorizing outcomes into high and low-risk groups for diabetes progression in our study.

Values which are 140 and above are categorized into class 'high' and values which are less than 140 falls into class 'low', Thus, it becomes a balanced classification problem with each category consisting of equal number of data points.

IV. SYSTEM ARCHITECTURE

System architecture (Fig. 3) consists of four main blocks, which involve i) Input data consisting of body parameters and blood serum measurements ii) preprocessing of data iii) application of optimized ML algorithm and iv) the classification of data to any of the two classes. After these processes, the classification efficiency and accuracy of the model are analyzed. The methods used for each of these stages are depicted under section V.

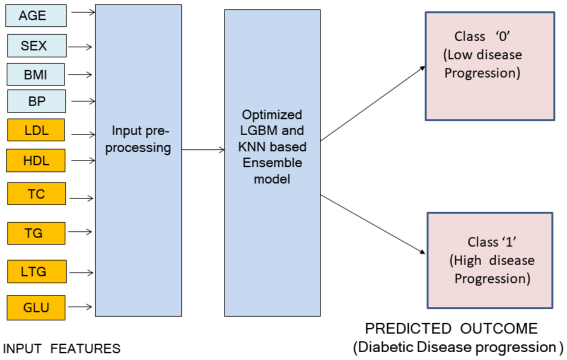

AGE
SEX
BMI
BP
LDL
HDL
TC
TG
LTG
GLU
INPUT FEATURES
Input pre-processing
Optimized LGBM and KNN based Ensemble model
Class '0' (Low disease Progression)
Class '1' (High disease Progression)
PREDICTED OUTCOME
(Diabetic Disease progression)

FIGURE 3. **System architecture of the proposed model.**

V. PROPOSED METHOD

The proposed method indicates the step-by-step analysis adapted in the study to optimize and come up with the final model. First step is the usage of right techniques for preprocessing the data, to make sure that the data is optimized to be used for further ML based analysis (part A). After performing the same, the efficient LightGBM algorithm was used with random search method for hyperparameter tuning, and the performance parameters were analyzed (section B). It produced 75% accuracy with AUC of ROC of 81.5%. Now, in order to optimize further, ensemble model of LightGBM with KNN with voting classifier was tested with grid search method for hyper parameter tuning (part C). This ensemble has resulted in 75% accuracy with AUC of ROC of 83.2%, which represents good classification efficiency. Comparison of the final optimized model with other state of the art classification models, including other ensemble models is performed, which ensured the supremacy of the model. In order to verify the robustness of the model, validation of the technique was also carried out with another diabetic dataset. The evaluation matrices showed promising results (Section D). Finally, the experimentation and implementation in Azure Cloud using Azure ML is discussed (section E), which adds to the optimization process in terms of scalability and resource usage.

A. PREPROCESSING

Preprocessing of data varies from model to model according to the requirement. Here the methods indicated in subsections I,II, III and IV are used to prepare the dataset to an optimum level to be used for further processing.

1) FINDING DISTRIBUTION AND DATA CLEANING

In this step, the distribution of outcome is visualized first. The outcome is distributed as 50% of high disease progression and remaining 50% as low disease progression. The distribution of various features with respect to the outcome has been analyzed and the obtained violin plots of some features are depicted in Fig. 4, 5, 6 and 7, which show the distribution of features Glu, Age, BP, TC, LDL, HDL, BMI and SEX against the outcome respectively.

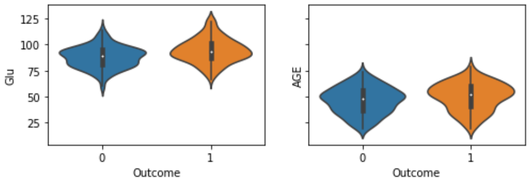


FIGURE 4. **Outcome versus Glucose and AGE.**

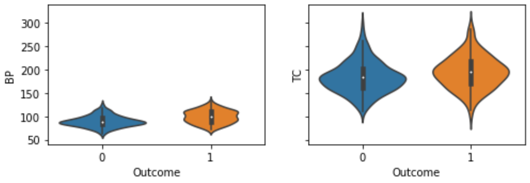


FIGURE 5. **Outcome versus BP and TC.**

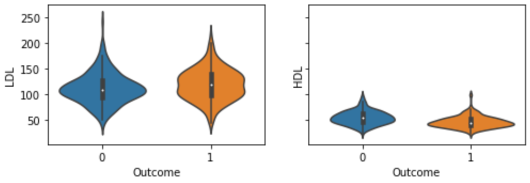


FIGURE 6. **Outcome versus LDL and HDL.**

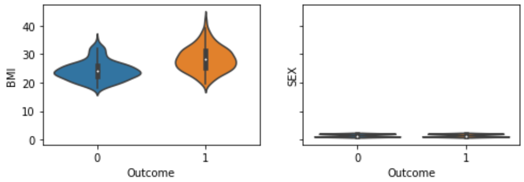


FIGURE 7. **Outcome versus BMI and Sex.**

The analysis reveals that individuals in high-risk class ‘1’ tend to be around 60 years old, while low-risk class ‘0’ includes a broader age range below 50. Class ‘1’ has more instances with BP levels above 100, whereas class ‘0’ mostly has levels below 100. Class ‘1’ individuals tend to have TC levels around 200, whereas class ‘0’ individuals exhibit lower levels. HDL levels in class ‘0’ are mainly distributed at 50 and above, while LDL levels are higher in class ‘1’. BMI patterns show class ‘1’ individuals tending towards overweight (around 30), while class ‘0’ individuals stay within the normal range (25 or below). Gender is coded as ‘1’ for males and ‘2’ for females. Once this distribution is examined, next step is to perform data cleaning, which checks for missing values, irrelevance or outliers. Here, dataset is clean and hence retained as such. The next process is data scaling.

2) SCALING OF DATA

Standard scaler is used to scale the data. It removes the mean and scales to unit variance, thereby standardizing the features. By computing the necessary statistics on the samples in the set, features are centered and scaled independently. Mean and standard deviation are then stored to be used later on the data. Standardization of a dataset is required because if the individual features are not normally distributed, it will not produce the expected results [34]. Next, the correlation among various features are analysed.

3) FEATURE ANALYSIS

The correlation of features indicates the similarity measures among the features. The correlation plot of all the features with each other is shown in Fig. 8.

Referring to Fig. 8, pure yellow represents 100% correlation, which is obtained diagonally among the same features, while the least correlated features are presented in dark blue shade. Light shades denote good correlation, while dark shades of blue and green denote less correlation. The remaining correlations are represented as the shades in between. The correlation scores are also displayed. Here we can understand that features such as BMI, BP and LTG are more correlated to the outcome than features like HDL and AGE of people. If the correlation is very high between the features, typically in the range of 0.9, multicollinearity may arise. This indicates that when two variables are strongly correlated, their presence can lead to redundancy, which negatively impacts the overall performance of the model. Conversely, if a variable has a very low correlation with others, it suggests that removing it will likely have minimal effect on model performance. In this case, LDL and TC

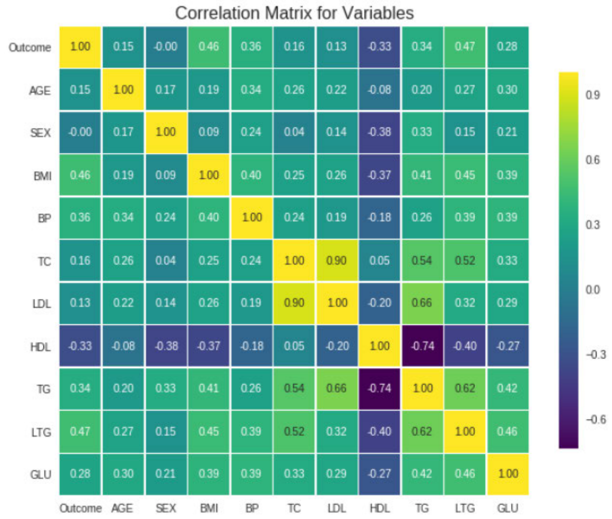


FIGURE 8. Correlation plot of the features.

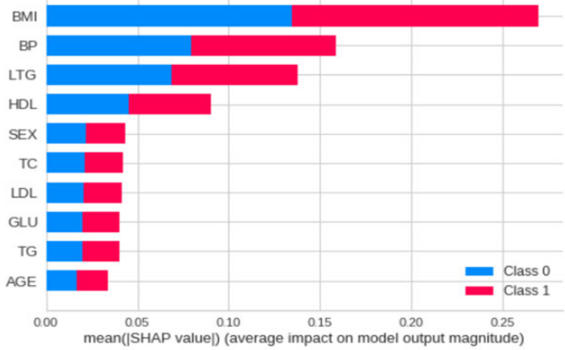


FIGURE 9. Impact of features on the outcome using SHAP analysis.

are highly correlated with value 0.9, which could lead to multicollinearity. Additionally, LDL shows lower correlation with other parameters. Therefore, eliminating LDL might not significantly affect the model's performance compared to TC, which has a stronger correlation with the other factors. Moreover, HDL is less correlated with the outcome as well.

But the disadvantage of correlation technique is that, the correlation scores can capture the relationship if the features are linearly related. When the features are not linearly related correlation doesn't help. Hence, in order to identify the influence of each feature on the outcome, SHapley Additive exPlanations (SHAP) analysis was conducted using Random Forest classifier. The SHAP values are calculated by analysis of predictions, without and with a feature being present. This activity is performed for each feature and each record in the dataset in an iterative fashion. Each feature is then provided with an importance value for every prediction. Hence it will provide an explanation on the impact of each feature on the outcome [35]. The results of SHAP analysis on our dataset is depicted in Fig. 9. Here, it indicates that outcome is more impacted by features such as BMI, BP and LTG for both the classes than other features such as AGE or TG. This brings up explainability angle to the model. SHAP analysis helps in identifying which features are aligning more towards the outcome of diabetic prediction in comparison with the other features considered. This helps in interpreting the model, which can add more value, especially in medical field. Here, all the features are retained as the model performance was unaffected by feature reduction.

4) CROSS VALIDATION

Cross validation is performed on the data by splitting the dataset in different folds; to be used as training and testing set. In k fold cross validation method, data is divided into k folds; in which $k - 1$ folds are used for training and one fold is used for testing the data [36]. In every iteration, the fold which is used for testing changes and hence the training data also varies. Testing and training is performed on all dataset belonging to different folds, by the end of k iterations and the average score is taken [36]. Hence it results in unbiased accuracy score. After applying all the preprocessing techniques, performance analysis is conducted by applying LightGBM with k-fold cross validation using random search and the results are compared with the results obtained after applying proposed Ensemble method using Grid search. By applying different values for k; it was found that 10-fold cross validation has given maximum results.

B. PERFORMANCE ANALYSIS OF LIGHTGBM USING RANDOM SEARCH

The performance analysis of LightGBM with random search method is presented here. The parameters used to estimate the results of the method on the dataset are depicted in Table 1.

TABLE 1. Details of parameters used for LightGBM method.

Number of estimators	100 - 2000
Number of leaves	6 - 50
Minimum child samples	100 - 500
Maximum depth	7
Learning rate	0.01 – 0.4
Number of iterations	300
Early stopping rounds	100

Referring to Table 1, the number of estimators represent the number of trees/rounds and the number of leaves indicates total leaves in the full tree. The number of minimum child sample, represents the minimum number of data in a leaf. The learning rate depicts the impact of each tree on the final outcome. Number of iteration stands for number of combinations of parameters. Early stopping rounds are used to stop the iterations, in case of no further improvement in learning. The model performance scores, ROC and Precision Recall curve of LightGBM method using random search with 10-fold cross validation is shown in Fig. 10, Fig. 11 and the detailed performance report is presented in Table 2.

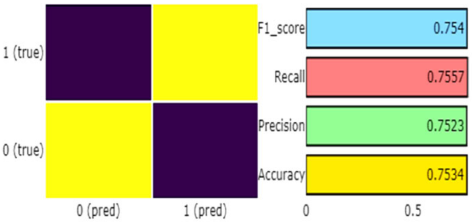


FIGURE 10. Performance report of LightGBM method using random search.

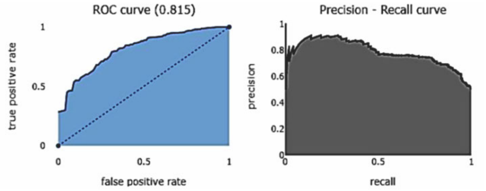
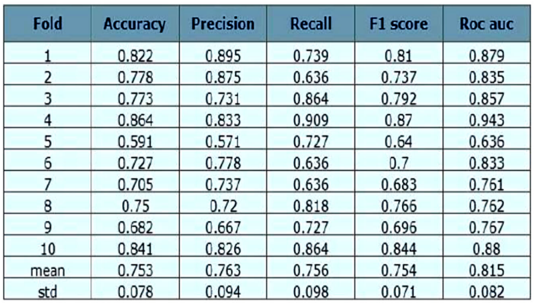


FIGURE 11. ROC and PR curve of LightGBM method using random search.

TABLE 2. Detailed performance report of LightGBM method using random search.



Fold	Accuracy	Precision	Recall	F1 score	Roc auc
1	0.822	0.895	0.739	0.81	0.879
2	0.778	0.875	0.636	0.737	0.835
3	0.773	0.731	0.864	0.792	0.857
4	0.864	0.833	0.909	0.87	0.943
5	0.591	0.571	0.727	0.64	0.636
6	0.727	0.778	0.636	0.7	0.833
7	0.705	0.737	0.636	0.683	0.761
8	0.75	0.72	0.818	0.766	0.762
9	0.682	0.667	0.727	0.696	0.767
10	0.841	0.826	0.864	0.844	0.88
mean	0.753	0.763	0.756	0.754	0.815
std	0.078	0.094	0.098	0.071	0.082

Table 2 portrays performance parameters such as accuracy, precision, F1 score and area under ROC curve obtained after each iteration. The mean values of accuracy is 75.3%, Area Under Curve (AUC) of ROC is 81.5%, the precision score is 76.3%, while mean of Recall and F1 scores are 75.6% and 75.4% respectively. The ROC curve of Figure 11, signifies the relationship between true positive rate against false positive rate, while the Precision Recall (PR) curve depicts values of precision against various threshold values of Recall.

C. PERFORMANCE ANALYSIS OF THE PROPOSED OPTIMIZED ENSEMBLE MODEL USING GRID SEARCH

After performing the accuracy analysis of LightGBM using random search; experiment was conducted by incorporating LightGBM classifier and KNN classifier into an Ensemble voting classifier. Using Grid search method, the best fit value for K in KNN classifier was found out as 23. This value of K was used while performing voting by the voting classifier. The details of the parameters used for the voting classifier is indicated in Table 3. For a given input information of a patient, the steps involved in predicting the final probability using the Soft Voting Classifier for a LightGBM and KNN based ensemble are as follows:

TABLE 3. Details of parameters used for voting classifier.

Estimators	KNN, LightGBM
Voting	Soft Voting
Weights	1:1

Let the probability of LightGBM prediction for the output to fall in class ci, be represented as expression 3.

$$PLGBM(y = ci|x) \tag{3}$$

where x represents the given input information for a patient, and y represents the outcome. ci depicts the class, where i can take values of either '0' or '1'.

Similarly, Let the probability of KNN prediction of the output to fall in class ci for a given input be represented as expression 4.

$$PKNN(y = ci|x) \tag{4}$$

where x represents the given input information for a patient, and y denotes the outcome. Now, the final combined probability using soft voting method $Pv(y = ci|x)$, for the outcome y to be in class ci for the x input information of a patient, can be implied as equation 5.

$$Pv(y = ci|x) = \frac{PLGBM(y = ci|x) + PKNN(y = ci|x)}{2} \tag{5}$$

From equation 5, we can understand that soft voting method combines the probabilities of both models in equal proportion. Based on which class has resulted in higher average probability score, the voting classifier chooses the final class, for the given input features of a patient. The performance report of the optimized model using ensemble method is presented in Fig. 12, Fig. 13 and Table 4.

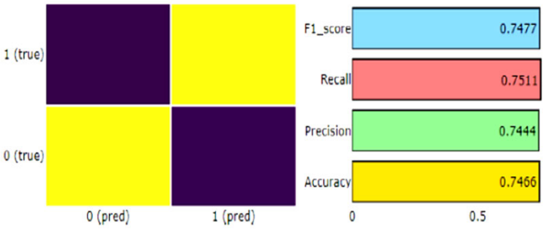


FIGURE 12. Performance report of the proposed model.

Referring to Fig. 12, Fig. 13 and Table 4, we can see that the accuracy score is 74.6%, which is almost the same as the score obtained by applying LightGBM alone. The AUC of the ROC obtained for the ensemble model is 83.2%. It shows an improvement of 1.7% from the result obtained when LightGBM was applied (81.5%) individually.

The same analysis has been carried out by changing the value of k into different folds. But 10-fold cross validation has given the maximum score for this model.

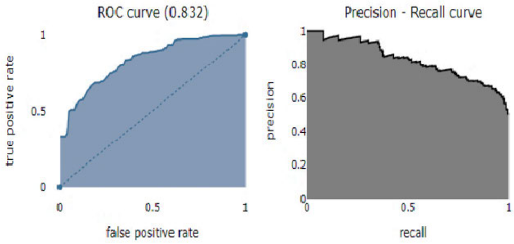
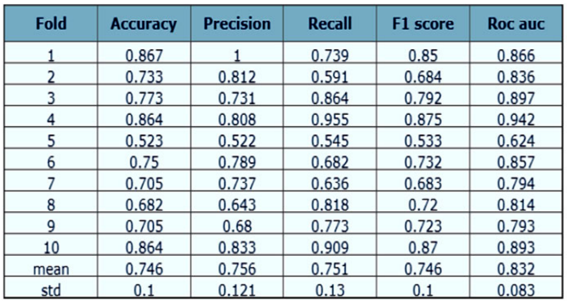


FIGURE 13. ROC and PR curves of the proposed model.

TABLE 4. Detailed performance report of the proposed model.



Fold	Accuracy	Precision	Recall	F1 score	Roc auc
1	0.867	1	0.739	0.85	0.866
2	0.733	0.812	0.591	0.684	0.836
3	0.773	0.731	0.864	0.792	0.897
4	0.864	0.808	0.955	0.875	0.942
5	0.523	0.522	0.545	0.533	0.624
6	0.75	0.789	0.682	0.732	0.857
7	0.705	0.737	0.636	0.683	0.794
8	0.682	0.643	0.818	0.72	0.814
9	0.705	0.68	0.773	0.723	0.793
10	0.864	0.833	0.909	0.87	0.893
mean	0.746	0.756	0.751	0.746	0.832
std	0.1	0.121	0.13	0.1	0.083

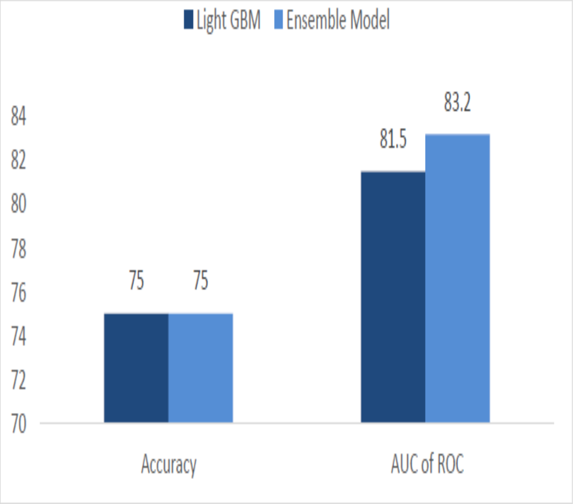


FIGURE 14. Performance report of LightGBM with proposed optimized model.

D. PERFORMANCE COMPARISON OF OPTIMIZED ENSEMBLE MODEL WITH OTHER MODELS AND RESULT VALIDATION

Firslty, comparison chart between the individual LightGBM model explained in section B and the proposed Ensemble model described in section C is shown in Fig. 14. From Fig. 14, it is clear that the proposed Ensemble model outperforms the LightGBM in terms of AUC of ROC by 1.7%, even though, accuracy values remains the same for both the models. The proposed model was also compared with the other state of the art classification algorithms, such as LR, KNN, CART, SVM, Naive Bayes and a sequential Neural Network model with Adam optimization. The resulting plot of comparison is represented in Fig. 15.

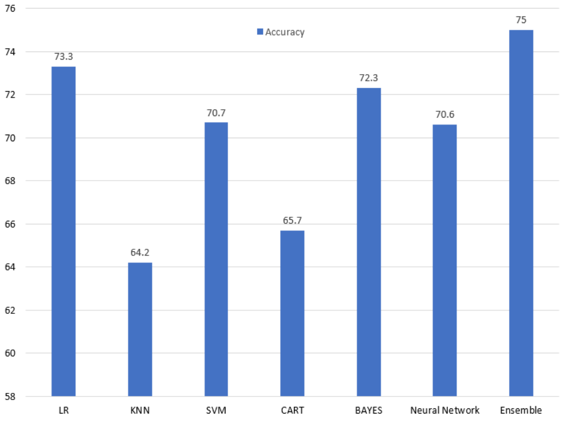


FIGURE 15. Performance comparison of Proposed model with other classification models.

From figure 15, it can be inferred that the proposed Ensemble method has 75% accuracy, indicating its supremacy. compared to the other models which produced lesser scores.

Article [37] explores the use of XGboost technique in type-1 diabetic prediction. Studies [38], [39], [40] investigate different tree-based ensembles for blood glucose prediction. Bagging and Boosting using decision trees have been experimented in these papers. Adaboost, Random Forest and other algorithms have also been tested on various data. By analyzing such high performing algorithms from the literature, a comparative study of the performance of the proposed model with the accuracy of the other ensemble models is performed and results are represented in Fig. 16. Referring to the Fig. 16, the proposed model (Ensemble Model), is compared with other tree-based boosting and bagging ensembles such as Adaboost, Bagging with KNN, Catboost, Random Forest, Xgboost etc. It was found that XGboost has also produced the same accuracy as the proposed Ensemble model. Hence to differentiate between the two models, the AUC of ROC of both the methods were also compared. The resulting graph is depicted in Fig.17. From the Fig. 17, it is observed that XGboost and proposed model have same accuracy, whereas the AUC of ROC shows difference, indicating that the proposed model produced better classification efficiency (83,2%) compared to XGBoost(75%).

In order to verify the performance of the proposed ensemble model with the latest meta heuristic optimization methods, the model was also optimized with Bayesian Optimization (BO) technique as well as Genetic Algorithm (GA). Bayesian Optimization (BO) is a good approach for optimization, in scenarios where there is a constraint on the computational resource [41]. Bayesian Optimization uses concepts of Bayes theorem for guiding the search in hyperparameter optimization by creating a probabilistic

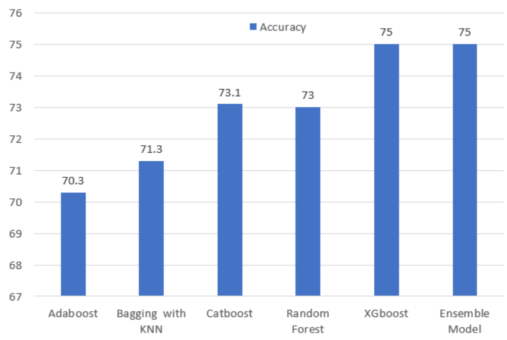


FIGURE 16. Performance comparison of proposed model with other Ensemble techniques.

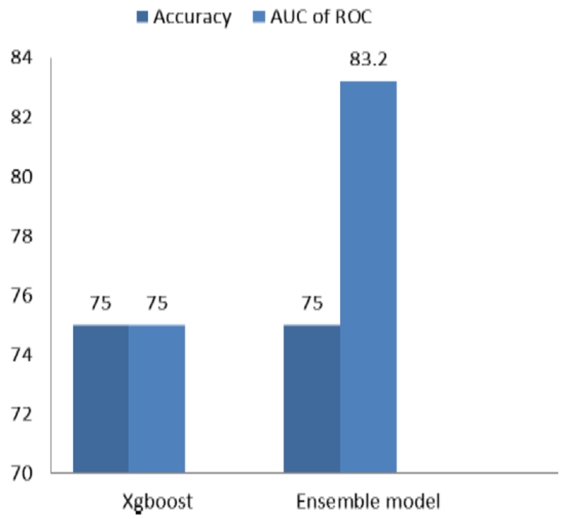


FIGURE 17. Performance comparison of XGboost with proposed model in terms of AUC of ROC and accuracy.

model named surrogate model, that imitates the objective function. It searches as much space as possible, when configuration space is large and provides best results in fewer trials. The Genetic Algorithm is an evolutionary algorithm which assumes each candidate solution as a chromosome [42]. The genes represent the hyperparameters. Hence, the number of genes in each chromosome denotes the number of hyperparameters. GA chooses some set of chromosomes and best ones are shortlisted based on the fitness function. Afterwards, mutation and crossover is applied to provide the best solution. The GA can make the solution better over time in cases where the solution is stored in memory. But it demands a lot of memory and computational resources.

Considering the advantages of the methods, the techniques are tested with our ensemble and the performance parameters are evaluated. These optimization methods produced comparable results, even though, the proposed Grid search (GS) method with ensemble resulted in better classification efficiency in terms of AUC of ROC (AUC) in small margin. The final results obtained with all the methods are represented in Table 5.

TABLE 5. Performance comparison of proposed optimization technique with BO and GA.

Optimization	Accuracy	Precision	Recall	F1-Score	AUC
B O	74	75.4	72.4	72.9	80.9
GA	73.7	75.3	72	73.2	82.3
GS	74.6	75.6	75.1	74.6	83.2

We can observe that, for BO, the mean accuracy is 74%, precision is 75.4%, Recall is 72.4%, F1 score is 72.9% and AUC of ROC measures to 80.9%. GA produced mean accuracy of 73.7%, precision of 75.3% While recall and F1 score are 72% and 73.2% respectively. The ROC of AUC obtained is 82.3%. Comparing it to the proposed GS technique, GS has resulted in slightly better overall results, eventhough the meta heuristic methods have produced comparable results.

To verify if the algorithm overfits the data and to understand its robustness and generaliziability, the algorithm is also validated with another dataset which is an 'Early stage diabetes risk prediction dataset' contributed in 2020 in UCI ML Repository. The link to the dataset is https://doi.org/10.24432/C5VG8H. This dataset contains the sign and symptom data of 520 patients, who are newly diabetic or would be diabetic. The parameters considered are Age, Sex and some conditions in patients such as Occurrence of Polyuria, Polydipsia, Sudden weight loss, weakness, Polyphagia, Genital thrush, visual blurring, itching, irritability, delayed healing, partial paresis, muscle stiffness, Alopecia and obesity. Here, the outcome indicates if the diabetic risk of the patients are positive or negative.

The resulting confusion matrix and performance parameters are presented in Fig.18 and Table 6.

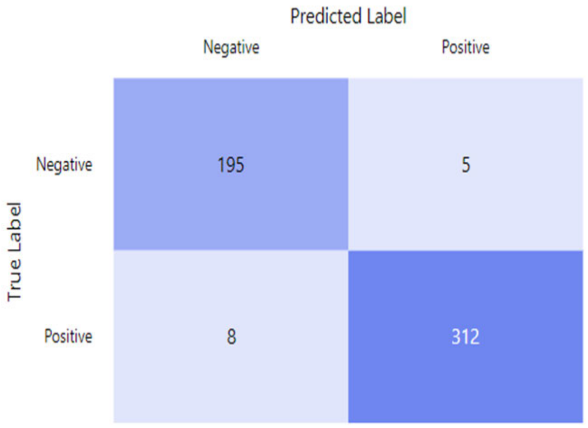


FIGURE 18. Confusion marix obtained for dataset 2 using proposed method.

TABLE 6. Performance report of the validation dataset.

Accuracy	AUC	Precision	F1-Score	Recall
97.5	99.5	97.5	97.6	97.5

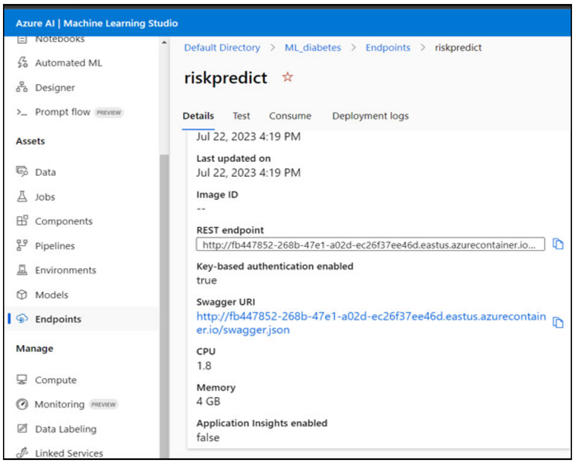
From Fig. 18 and Table 6, it can be inferred that the proposed algorithm has produced good performance scores, in the range of 97% and above. These experiments prove the capability of proposed Ensemble method in terms of all performance parameters.

E. CLOUD IMPLEMENTATION

In the context of organizations increasingly migrating their operations to the cloud for efficient management of storage, infrastructure and platforms, embedding ML models as web services within the cloud proves helpful. Prominent cloud providers offer services that eliminate concerns over hardware, software, networking, and scaling. Microsoft Azure, for instance, offers a platform to execute ML tasks and deploy them as web services or real-time endpoints [43]. In Azure ML Studio, the ML based experiments can be performed using Python, and the best model can be registered and then deployed. Deployment involves selecting a compute target, usually the same used for model experimentation. Upon successful deployment, an endpoint is created within Azure as a web service, accessible for testing the ML model's prediction service. The generated REST endpoint is a crucial outcome of a successful deployment process.

The steps used in Azure Machine learning Studio, for model training and deployment are as follows.

- Set up Azure Machine Learning workspace: After logging into the azure portal, a resource group is created, which will group all the resources that we use within the ML project. The workspace is the resource for azure machine learning activities, which provides a centralized place to view and manage the artifacts we create for our ML based projects. After signing into azure machine learning studio, a workspace is created with proper configuration, by inputting the details of subscription, resource group and Azure region. Following the selection of integrated Python based Jupyter Notebook, a compute instance is created, which is a pre-configured cloud-computing resource that we use to train, automate, manage, and track machine learning models. Here we have used Standard DS11_V2 which has 2 cores, 14 GB RAM and 20 GB storage. This is used to run Jupyter notebooks and Python scripts for the ML project in hand. Before jumping into the code, ML client is created as a handle to the workspace, which manages resources and jobs.
- Create the training script: The training script created handles the preprocessing of the data. It then consumes this data to train the model and returns the output model. MLflow platform is utilized to log the parameters and metrics during the pipeline run. Once the model is trained, the model file is saved and registered to the workspace. The registered model is then used in inferencing endpoints.
- Create a scalable compute cluster: A compute cluster is created which can be scaled up and down based on the requirements, that ensures, we utilize the right number of compute resources for the project. This helps in no wastage or lack of the compute resources.
- Create and run a command job: As the script is created for the ML task and a compute cluster is set up to run the script in the previous steps, commands are used to run the script. After this, the job is submitted.
- View the output of the training script: View the job in Azure Machine Learning studio. The tabs indicate various details like metrics, outputs etc. After the completion of the job, it registers the model in the workspace, indicated as the result of the training job performed.
- Deploy the newly-trained model as an endpoint: Once the output metrics obtained indicate good efficiency, deployment can be performed as a web service in the Azure cloud. After the endpoint is created, deploy the model with the entry script. Here, the optimized model with both progression and validation datasets are deployed as web service, that handles 100% of the incoming traffic. The models page on Azure Machine Learning studio, displays the latest version of the registered model. The endpoint generated for the model after successful deployment is represented in Fig. 19.



riskpredict ☆

Jul 22, 2023 4:19 PM

Last updated on
Jul 22, 2023 4:19 PM

Image ID
--

REST endpoint
http://fb447852-268b-47e1-a02d-ec26f37ee46d.eastus.azurecontainer.io...

Key-based authentication enabled
true

Swagger URI
http://fb447852-268b-47e1-a02d-ec26f37ee46d.eastus.azurecontainer.io/swagger.json

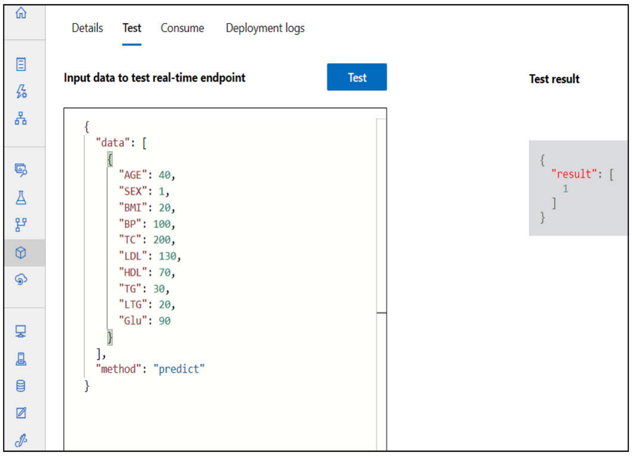
CPU
1.8

Memory
4 GB

Application Insights enabled
false

FIGURE 19. Sample of the rest endpoint generated after successful deployment of the model in Azure cloud.

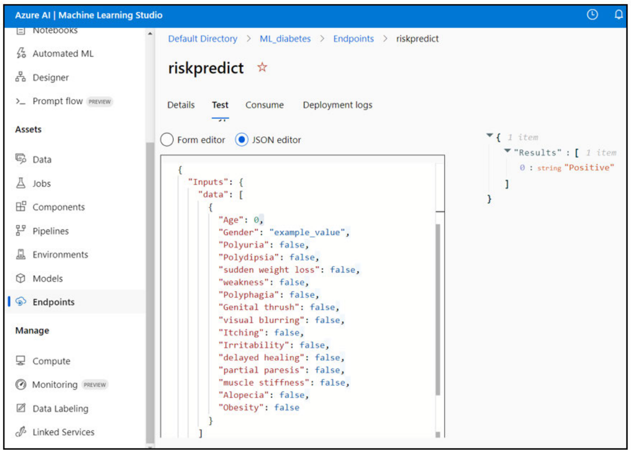
- Call the Azure Machine Learning endpoint for inferencing: After successful deployment of the model as a web service, we can test the prediction by running inference with it. A sample request file is built to test the prediction for a given set of input parameters. The prediction results of progression dataset and risk prediction dataset used for validation are depicted in Fig. 20 and Fig. 21 respectively.



```
{
  "data": [
    {
      "AGE": 40,
      "SEX": 1,
      "BMI": 20,
      "BP": 100,
      "TC": 200,
      "LDL": 130,
      "HDL": 70,
      "TG": 30,
      "LTG": 20,
      "Glu": 90
    }
  ],
  "method": "predict"
}
```

```
{
  "result": [
    1
  ]
}
```

FIGURE 20. Testing the prediction of the web service deployed in cloud for the progression dataset.



```
{
  "Inputs": {
    "data": [
      {
        "Age": 0,
        "Gender": "example_value",
        "Polyuria": false,
        "Polydipsia": false,
        "sudden weight loss": false,
        "weakness": false,
        "Polyphagia": false,
        "Genital thrush": false,
        "visual blurring": false,
        "Itching": false,
        "Irritability": false,
        "delayed healing": false,
        "partial paresis": false,
        "muscle stiffness": false,
        "Alopecia": false,
        "Obesity": false
      }
    ]
```

```
{ 1 item
  "Results" : [ 1 item
    0 : string "Positive"
  ]
}
```

FIGURE 21. Testing the prediction of the web service deployed in cloud for the risk prediction dataset.

TABLE 7. Configuration details of Azure ML experiment.

Configuration Parameter	Name/Value	Details
Compute Type	Azure ML Compute Cluster	Managed cluster for training and deploying models.
VM Type	Standard_DS11_v2	Standard Virtual Machine.
Number of VMs	1-2	One VM sufficient for simple models. Scaling up to 2 nodes are given to be used, if required.
Total CPU Cores	2	More cores can be used for faster processing for large datasets
Memory Size	14 GB	Ensures sufficient memory for data processing.
Storage Size	20 GB (Standard SSD)	Sufficient for medium datasets, models and outputs.
Networking	Standard VNet Configuration	Ensures secure access for data and model management.
Duration for model training	50 seconds	For training small to medium models on 442 data points.

The configuration details of resources used to run the experiment in Azure cloud is depicted in Table 7.

TABLE 8. Comparison of azure machine learning with on premise machine learning.

Parameter	Azure Machine Learning	On premise Machine Learning
Computational efficiency	Offers upto **90%** faster training times for larger datasets.	Training time will be 2 to 5 times longer for large models.
Latency	The response time is **<100 ms** in most cases.	latency can be **>500 ms,** for local deployment.
Cost Efficiency	It can reduce costs by **30-40%** compared to maintaining local infrastructure.	It results in **20-30% higher costs** over time.
Model training time	Azure ML makes use of powerful GPU/TPU resources, which reduces training time from hours to **minutes**.	Training on local CPU may take **hours to days**, depending on the model complexity and size of the dataset.
Collaboration	It increases team productivity by **20-25%**.	It decreases productivity by **10-15%**.
Scalability	It can handle workload spikes efficiently.	It can take **weeks to months** to upgrade the infrastructure.
Security and compliance	It reduces compliance risk by **up to 60%**.	It leads to potential compliance risks and incurs higher costs for security measures.
Automatic Updates	It saves **10-15%** in maintenance costs.	It leads to **increased downtime** and resource allocation time.

Based on the number of data points in the dataset and the complexity of the model training, configuration mentioned in Table 7, was chosen in Azure Machine Learning Studio. We have used Standard DS11_v2 Virtual Machine with 14GB RAM for Model training. It has taken 50 seconds for the training while on a normal PC it can take up to 2 minutes. Major difference appears, especially when the dataset is huge, and algorithm has complex steps. In such scenarios, the auto scaling capability of cloud makes huge difference since it uses the resources as per the requirement. When we implement the same on premise, we will have to take care of the infrastructure required and the networking aspect as well. Based on the compute resources we use, depending upon the complexity of the model, there are different pricing plans of using Azure Machine Learning studio. We have utilized Azure subscription with free credits, to train and deploy the model. The consumed cost depends upon how long we use the compute resource switched on. We have used compute resource, Standard DS11_V2, with 2 vCPUs and 14 GB RAM, priced at $0.1850/hour. Azure ML is offered in 2 tiers which are free and standard. Under free tier, 1 hour of experiment time is allowed without Azure subscription. Maximum storage space of 10GB with 1 node is permitted as per this plan. Under standard tier, $9.99 per ML studio workspace per month, with $1 per studio experimentation per hour is provisioned, with Azure subscription. Maximum experiment duration upto 7 days per experiment, with a maximum of 24 hours per module is permitted under this tier [44]. It offers unlimited storage space with multiple nodes provided for execution. We can utilize the cost analysis tool in Azure to monitor the costs of our resources. For Implementation on real time system, we may use Azure IoT hub with proper license and legal permission from the medical field.

Comparison of using Azure ML in Azure cloud against the use of on premise ML, with respect to various factors in quantitative terms is presented in Table 8. We can understand that Cloud implementation is much faster and scalable aiding to improved computational and cost efficiency. It shows superiority in terms of latency, collaboration, security etc. as well. The deployment in Cloud, globally offers low latency, which reduces to less than 100ms of response time. Whereas a local deployment will cause latency of more than 500ms. When the data under consideration is huge, we need more than the local CPU for efficient model training and deployments. In such situations, the cloud provides flexibility for scaling. In Cloud, we get access to the latest GPUs or even TPUs which is difficult to manage and maintain on premise. Azure provides security and privacy as well with respect to the data used. The resource demand of ML experiments brings it close to cloud computing; as it requires a lot of processing power, data storage and many servers to work together on an algorithm, especially if the problem at hand is complex with respect to the data and algorithm. In such scenario, using cloud computing, we can spin up the number of severs to work on the algorithm and can later scale it down, when not required. Hence, implementation on cloud will make sure that the resources are utilized as per the dynamic requirement. Therefore, cloud computing provides superior cost efficiency through lower initial investments, scalability, reduced maintenance, optimized resource use, and the offloading of operational burdens to the service provider. Whereas, on premise services will have to deal with huge initial investment and maintenance cost with less flexibility in terms of resource usage, scalability etc. Leveraging the capabilities of Azure Machine Learning, potentially helps real time implementation scenarios in a Health IoT based ecosystem.

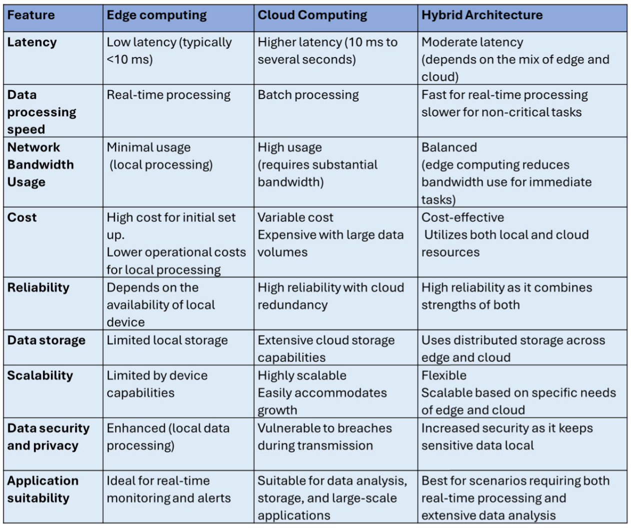
VI. RESULT DISCUSSION

We have seen the promising results obtained for the model as well as the advantages of using cloud. The implementation of web service in cloud if integrated in Health IoT system will have a lot of benefits compared to the traditional methods in terms of overall treatment strategy and patient satisfaction [45], [46], [47], [48] as enumerated below.

- It shows significant improvement in high risk patient identification compared to traditional method of identifying the disease.
- Increase in treatment adherence rate for the patients were identified compared to the traditional method.
- Major decline in average hospitalization per year was detected in HIoT implementation, compared to traditional method.
- Real time implementation in Health IoT system results in substantial decline in complications.
- There is major rise in patient engagement score in HIoT compared to traditional method.
- Cost of care has reduced per year which adds to improvement in affordability of treatments.
- There is less time required on patient monitoring compared to traditional method.
- Rise in patient satisfaction rate in HIoT method is commendable against the rate in traditional approach.

However, the current studies indicate the issues with respect to the resources and the possibility of using edge computing and its advantages. Study [49], throws light on health based IoT architecture, Internet of Medical Things (IoMT), which contributes to the remote monitoring of patients. The paper discusses glucose prediction for T1D patients with respect to the IOMT scenario. Constrained devices have limited computational power, which makes it difficult to run those ML algorithms directly. Here concept of edge computing comes into play, where light weight ML algorithms can be performed on those devices [49]. This is a valid solution to avoid resource constraints. However, for scenarios where large computational requirements are prevalent in the cases of complex models, cloud computing can be leveraged. Eventhough, the devices collecting values are of less computational power, we can utilize the Cloud for performing the computations required for Machine Learning. Hence usage of ML predictions in cloud will reduce the burden of performing the computation at the device level, if resources are available. So, by balancing the capabilities of both Edge and cloud computing, a hybrid architecture offers a lot of flexibility. Considering these aspects, study has been conducted and the results are presented in Table 9.

TABLE 9. Comparison of cloud computing with edge computing and hybrid architecture.



Feature	Edge computing	Cloud Computing	Hybrid Architecture
Latency	Low latency (typically <10 ms)	Higher latency (10 ms to several seconds)	Moderate latency (depends on the mix of edge and cloud)
Data processing speed	Real-time processing	Batch processing	Fast for real-time processing slower for non-critical tasks
Network Bandwidth Usage	Minimal usage (local processing)	High usage (requires substantial bandwidth)	Balanced (edge computing reduces bandwidth use for immediate tasks)
Cost	High cost for initial set up. Lower operational costs for local processing	Variable cost Expensive with large data volumes	Cost-effective Utilizes both local and cloud resources
Reliability	Depends on the availability of local device	High reliability with cloud redundancy	High reliability as it combines strengths of both
Data storage	Limited local storage	Extensive cloud storage capabilities	Uses distributed storage across edge and cloud
Scalability	Limited by device capabilities	Highly scalable Easily accommodates growth	Flexible Scalable based on specific needs of edge and cloud
Data security and privacy	Enhanced (local data processing)	Vulnerable to breaches during transmission	Increased security as it keeps sensitive data local
Application suitability	Ideal for real-time monitoring and alerts	Suitable for data analysis, storage, and large-scale applications	Best for scenarios requiring both real-time processing and extensive data analysis

In our work, we have analyzed cloud implementation. However, as per Table 9, hybrid architecture looks promising too. It can be leveraged while implementing real time models. According to Table 9, we can observe that the hybrid architecture works well with all scenarios. In situations where devices have to perform some computations at a local level, we can utilize edge computing. Whereas, in places where extensive data processing and computations are required, cloud implementation will be beneficial. Hence a hybrid implementation approach will take care of both the cases.

Edge computing and hybrid cloud architectures are transforming the benefits for healthcare, but their adoption faces significant challenges, including interoperability, network reliability, cost constraints, and regulatory hurdles [50]. Ensuring compatibility across diverse technologies, addressing network issues in remote areas, and managing the financial and resource demands of implementing these technologies are key obstacles. In addition to that, maintaining compliance with strict data privacy regulations and adapting to evolving standards presents challenges. Strategies such as adopting open standards, enhancing network reliability, and leveraging phased implementation can help overcome these obstacles. The growing integration of 5th Generation of mobile cellular network (5G) and AI technology, further enhances healthcare systems, driving innovations in diagnostics, personalized treatment, and improved healthcare delivery. With the right strategies, these technologies can revolutionize healthcare by creating a more efficient, secure, and patient-centric ecosystem.

Even though the blood serum measurements considered in the study may have indirect relationship with the genetics of a person, genetic factors and other comorbidity were not considered directly in the study. But according to the studies, these factors influence diabetes. The risk of developing type 2 diabetes (T2D) is 40% if one parent has T2D and 70% if both parents have the disease. First-degree relatives of people with T2D have about three times more probability of developing the disease. Research has shown that the people with family history of diabetes and related complications are more likely to develop diabetes and related complications. Similarly Some diseases and infections can also cause diabetes. Pancreatic damage, diseases that affect the pancreas, such as pancreatitis, pancreatic cancer, fibrosis or hemochromatosis can lead to diabetes, as a damage in pancreas will lead to non-production of insulin which is required for regulating blood sugar levels. If pancreas must be removed, it cannot produce insulin at all. When a person has an illness or infection, body releases more glucose into the blood for defense. This will cause increased blood sugar levels. Some medicines which are used to treat organ transplants can lead to diabetes or worsen the diabetic condition. Lifestyle and habits can have a significant impact on the risk of developing diabetes and its management [51]. Sedentary lifestyle and increased body weight can cause diabetes. Being active helps in maintaining a healthy weight. It improves the cell's ability to utilize blood sugar. It also helps to manage diabetes of people who are already affected by the disease. Diet is also a factor which can influence the chances of T2D diabetes in patients. Skipping meals or eating large meals at irregular time intervals can cause increase in blood sugar levels. Getting enough sleep can help manage diabetes. Smoking can also influence the chances of developing diabetes, as the smoke contains hazardous chemicals and carcinogens which is harmful for the body's organs. Drinking too much alcohol can make it harder to control blood sugar levels.

The performance of our models can be improved if these factors of patients can be considered. As the public data we have considered does not contain these factors, currently we have analyzed the data without considering comorbidity, other diseases, infections or lifestyle factors. But the blood serum measurements can indirectly point to these factors to a certain level. We have considered BMI in our study which may indicate sedentary lifestyle and lack of physical activity indirectly. The validation data which we have used in the study have considered some illness or infections of patients such as occurrence of Polyuria, Polydipsia, Sudden weight loss, weakness, Polyphagia, Genital thrush, visual blurring, itching, irritability, delayed healing, partial paresis, muscle stiffness and Alopecia related to diabetes, along with other factors. According to recent studies, MicroRNAs (miRNAs) are identified as crucial regulators of gene expression in both healthy and diseased individuals, with significant potential for improving the diagnosis of Type 2 Diabetes Mellitus (T2D) and its comorbidities. In study [52] it was shown that miRNAs like hsa-mir-1-3p, hsa-mir-16-5p, and hsa-mir-34a-5p can distinguish T2D samples from normal ones effectively. Overall, it will be beneficial to add lifestyle factors, comorbidity, infections and genetic factors to improve the results of our work. Section VII concludes the paper and incorporates future direction.

VII. CONCLUSION AND FUTURE WORK

The combination of the fast LightGBM algorithm with KNN in a voting classifier, with precise optimization, has yielded promising results, outperforming other models. The ensemble achieved an 83.2% AUC in the ROC analysis, showcasing its classification efficiency. The model's accuracy reached 75%. Utilizing 10-fold cross-validation, grid search method and optimal parameters provided a good average performance assessment. Notably, employing 10 input features on data from 442 patients distinguishes this method from literature that used fewer features and samples. Unlike immediate diabetic prediction, this model forecasts diabetic progression. The paper has extensively discussed the optimization as well as compared it with state of the art methods. The model's performance is validated using another risk prediction dataset, which produced superior results as well. The cloud implementation incorporates capabilities of cloud computing and enables the model's integration into smart healthcare systems, which helps in monitoring of patients remotely and providing medical advice in a fast manner. Comparison with latest developments in Edge computing and hybrid architecture has also been performed. However, inclusion of factors like genetics, exercise, infections or other comorbidity of patients are not considered here, which can be incorporated in future study. In alignment with the same, MicroRNA analysis for genetic feature extraction related to diabetic progression can be performed in future.

REFERENCES

[1] A. Oussous, F.-Z. Benjelloun, A. A. Lahcen, and S. Belfkih, "Big Data technologies: A survey," *J. King Saud Univ. -Comput. Inf. Sci.*, vol. 30, no. 4, pp. 431–448, 2018.

[2] V. K. Daliya and T. K. Ramesh, "Data interoperability enhancement of electronic health record data using a hybrid model," in *Proc. Int. Conf. Smart Syst. Inventive Technol. (ICSSIT)*, Nov. 2019, pp. 318–322.

[3] V. K. Daliya and T. K. Ramesh, "A survey on enhancing the interoperability aspect of IoT based systems," in *Proc. Int. Conf. Smart Technol. For Smart Nation (SmartTechCon)*, Aug. 2017, pp. 581–586.

[4] M. Marjani, F. Nasaruddin, A. Gani, A. Karim, I. A. T. Hashem, A. Siddiqa, and I. Yaqoob, "Big IoT data analytics: Architecture, opportunities, and open research challenges," *IEEE Access*, vol. 5, pp. 5247–5261, 2017.

[5] P. Xuan, C. Sun, T. Zhang, Y. Ye, T. Shen, and Y. Dong, "Gradient boosting decision tree-based method for predicting interactions between target genes and drugs," *J. Frontiers Genet.*, vol. 10, p. 459, May 2019.

[6] V. K. Daliya, T. K. Ramesh, and A. Shashikanth, "A machine learning based ensemble approach for predictive analysis of healthcare data," in *Proc. 2nd PhD Colloq. Ethically Driven Innov. Technol. Soc.*, Jul. 2020, pp. 1–2.

[7] V. K. Daliya and T. K. Ramesh, "A parameter tuned ensemble model for accurate prediction of diabetic progression," in *Proc. 3rd PhD Colloq. Ethically Driven Innov. Technol. Soc.*, Apr. 2021, pp. 1–2.

[8] O. Iparraguirre-Villanueva, K. Espinola-Linares, R. O. F. Castañeda, and M. Cabanillas-Carbonell, "Application of machine learning models for early detection and accurate classification of type 2 diabetes," *Diagnostics*, vol. 13, no. 14, p. 2383, Jul. 2023, doi: 10.3390/diagnostics13142383.

[9] A. Milad, N. I. M. Yusoff, S. A. Majeed, A. N. H. Ibrahim, M. A. Hassan, and A. S. B. Ali, "Using an Azure machine learning approach for flexible pavement maintenance," in *Proc. 16th IEEE Int. Colloq. Signal Process. Its Appl. (CSPA)*, Langkawi, Malaysia, Feb. 2020, pp. 146–150, doi: 10.1109/CSPA48992.2020.9068684.

[10] A. Z. Woldaregay, E. Årsand, S. Walderhaug, D. Albers, L. Mamykina, T. Botsis, and G. Hartvigsen, "Data-driven modeling and prediction of blood glucose dynamics: Machine learning applications in type 1 diabetes," *Artif. Intell. Med.*, vol. 98, pp. 109–134, Jul. 2019.

[11] H. M. Deberneh and I. Kim, "Prediction of type 2 diabetes based on machine learning algorithm," *Int. J. Environ. Res. Public Health*, vol. 18, no. 6, p. 3317, Mar. 2021.

[12] M. Makino, R. Yoshimoto, M. Ono, T. Itoko, T. Katsuki, A. Koseki, M. Kudo, K. Haida, J. Kuroda, R. Yanagiya, E. Saitoh, K. Hoshinaga, Y. Yuzawa, and A. Suzuki, "Artificial intelligence predicts the progression of diabetic kidney disease using big data machine learning," *Sci. Rep.*, vol. 9, no. 1, Aug. 2019, Art. no. 11862.

[13] A. Cahn and A. Shohen, *Prediction of Progression From Pre-diabetes to Diabetes- Development & Validation of a of a Machine Learning Model*. Hoboken, NJ, USA: Wiley, 2019.

[14] C. R. de Castro, L. Vigil, and B. Vargas, "Glucose time series complexity as a predictor of type 2 diabetes," in *Diabetes/Metabolism Research & Reviews*. Hoboken, NJ, USA: Wiley, 2017.

[15] N. J. Radcliffe, J. Seah, M. Clarke, R. J. MacIsaac, G. Jerums, and E. I. Ekinci, "Clinical predictive factors in diabetic kidney disease progression," *J. Diabetes Invest.*, vol. 8, no. 1, pp. 6–18, Jan. 2017.

[16] J. P. Anderson, J. R. Parikh, D. K. Shenfeld, V. Ivanov, C. Marks, B. W. Church, J. M. Laramie, J. Mardekian, B. A. Piper, R. J. Willke, and D. A. Rublee, "Reverse engineering and evaluation of prediction models for progression to type 2 diabetes: An application of machine learning using electronic health records," *J. Diabetes Sci. Technol.*, vol. 10, no. 1, pp. 6–18, Jan. 2016.

[17] E. Ferrannini and A. Mari, "Progression to diabetes on relatives of type 1 diabetic patients: Mechanism and mode of onset," *J. Diabetes*, vol. 59, no. 3, pp. 679–685, Mar. 2010.

[18] T. Hamdi, J. B. Ali, V. D. Costanzo, F. Fnaiech, E. Moreau, and J.-M. Ginoux, "Accurate prediction of continuous blood glucose based on support vector regression and differential evolution algorithm," *Biocybernetics Biomed. Eng.*, vol. 38, no. 2, pp. 362–372, 2018.

[19] H. N. Mhaskar, S. V. Pereverzyev, and M. D. van der Walt, "A deep learning approach to diabetic blood glucose prediction," *Frontiers Appl. Math. Statist.*, vol. 3, p. 14, Jul. 2017.

[20] K. Zarkogianni, K. Mitsis, E. Litsa, M.-T. Arredondo, G. Fico, A. Fioravanti, and K. S. Nikita, "Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring," *Med. Biol. Eng. Comput.*, vol. 53, no. 12, pp. 1333–1343, Dec. 2015.

[21] E. I. Georga, V. C. Protopappas, D. Polyzos, and D. I. Fotiadis, "Evaluation of short-term predictors of glucose concentration in type 1 diabetes combining feature ranking with regression models," *Med. Biol. Eng. Comput.*, vol. 53, no. 12, pp. 1305–1318, Dec. 2015.

[22] V. A. Kumari and R. Chitra, "Classification of diabetes disease using support vector machine," *Int. J. Eng. Res. Appl.*, vol. 3, no. 2, pp. 1797–1801, Apr. 2013.

[23] S. P. Nimmagadda, S. Yeruva, and R. Siempu, "Improved diabetes prediction model for predicting type-II diabetes," *J. Innov. Technol. Exploring Eng.*, vol. 8, no. 12, pp. 230–235, Oct. 2019.

[24] I. Rodríguez-Rodríguez, M. Campo-Valera, J.-V. Rodríguez, and W. L. Woo, "IoMT innovations in diabetes management: Predictive models using wearable data," *Expert Syst. Appl.*, vol. 238, Mar. 2024, Art. no. 121994, doi: 10.1016/j.eswa.2023.121994.

[25] A. Patria and Y. Patnaikb, "Random forest and stochastic gradient tree boosting based approach for the prediction of airfoil self-noise," in *Proc. Int. Conf. Inf. Commun. Technol. (ICICT)*, 2014, pp. 1–15.

[26] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. NIPS, Comput. Sci.*, 2017, pp. 1–9.

[27] V. K. Daliya and T. K. Ramesh, "Optimized stacking ensemble models for the prediction of diabetic progression," *Multimedia Tools Appl.*, vol. 82, no. 27, pp. 42901–42925, Apr. 10, 2023, doi: 10.1007/s11042-023-14858-4.

[28] Y. Wang, P.-F. Li, Y. Tian, J.-J. Ren, and J.-S. Li, "A shared decision-making system for diabetes medication choice utilizing electronic health record data," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 5, pp. 1280–1287, Sep. 2017.

[29] Y. Zhang, H. Zhang, J. Cai, and B. Yang, "A weighted voting classifier based on differential evolution," *Abstract Appl. Anal.*, vol. 2014, pp. 1–6, Jul. 2014.

[30] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–451, Apr. 2004, doi: 10.1214/009053604000000067.

[31] P. Susairaj, C. Snehalatha, A. Raghavan, A. Nanditha, R. Vinitha, K. Satheesh, D. G. Johnston, N. J. Wareham, and A. Ramachandran, "Cut-off value of random blood glucose among Asian Indians for preliminary screening of persons with prediabetes and undetected type 2 diabetes defined by the glycosylated haemoglobin criteria," *J. Diabetes Clinical Res.*, vol. 1, no. 2, pp. 53–58, 2019.

[32] S. Somannavar, A. Ganesan, M. Deepa, M. Datta, and V. Mohan, "Random capillary blood glucose cut points for diabetes and pre-diabetes derived from community-based opportunistic screening in India," *Diabetes Care*, vol. 32, no. 4, pp. 641–643, Apr. 2009, doi: 10.2337/dc08-0403.

[33] (2000). *Report Expert Committee Diagnosis Classification Diabetes Mellitus, Medscape—2000*. Accessed: Jul. 1, 2022. [Online]. Available: https://www.medscape.com/viewarticle/4126424

[34] L. Popova Zhuhadar and E. Thrasher, "Data analytics and its advantages for addressing the complexity of healthcare: A simulated Zika case study example," *Appl. Sci.*, vol. 9, no. 11, p. 2208, May 2019.

[35] F. Prendin, J. Pavan, G. Cappon, S. Del Favero, G. Sparacino, and A. Facchinetti, "The importance of interpreting machine learning models for blood glucose prediction in diabetes: An analysis using SHAP," *Sci. Rep.*, vol. 13, no. 1, Oct. 2023, Art. no. 16865, doi: 10.1038/s41598-023-44155-x.

[36] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-Fold cross validation in prediction error estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 569–575, Mar. 2010.

[37] G. Alfian, M. Syafrudin, J. Rhee, M. Anshari, M. Mustakim, and I. Fahrurrozi, "Blood glucose prediction model for type 1 diabetes based on extreme gradient boosting," in *Proc. Int. Conf. Inf. Technol. Digit. Appl.*, 2019, pp. 1–8.

[38] V. K. Daliya, T. K. Ramesh, and S.-B. Ko, "An optimised multivariable regression model for predictive analysis of diabetic disease progression," *IEEE Access*, vol. 9, pp. 99768–99780, 2021.

[39] B. A. Tama and K.-H. Rhee, "Tree-based classifier ensembles for early detection method of diabetes: An exploratory study," *Artif. Intell. Rev.*, vol. 51, no. 3, pp. 355–370, Mar. 2019.

[40] J. Liu, L. Wang, L. Zhang, Z. Zhang, and S. Zhang, "Predictive analytics for blood glucose concentration: An empirical study using the tree-based ensemble approach," *Library Hi Tech*, vol. 38, no. 4, pp. 835–858, Jul. 2020.

[41] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S.-H. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *J. Electron. Sci. Technol.*, vol. 17, pp. 26–40, Mar. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1674862X19300047

[42] M. Kaveh and M. S. Mesgari, "Application of meta-heuristic algorithms for training neural networks and deep learning architectures: A comprehensive review," *Neural Process. Lett.*, vol. 55, no. 4, pp. 4519–4622, Aug. 2023, doi: 10.1007/s11063-022-11055-6.

[43] M. B. Jamshidi, S. Roshani, J. Talla, M. S. Sharifi-Atashgah, S. Roshani, and Z. Peroutka, "Cloud-based machine learning techniques implemented by Microsoft Azure for designing power amplifiers," in *Proc. IEEE 12th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Dec. 2021, pp. 41–44.

[44] *Machine Learning Studio (Classic) Pricing*. Accessed: Nov. 30, 2024. [Online]. Available: https://azure.microsoft.com/en-in/pricing/details/machine-learning-studio/

[45] W. Rhmann, J. Khan, G. A. Khan, Z. Ashraf, B. Pandey, M. A. Khan, A. Ali, A. Ishrat, A. A. Alghamdi, B. Ahamad, and M. K. Shaik, "Comparative study of IoT- and AI-based computing disease detection approaches," *Data Sci. Manag.*, Jul. 2024, doi: 10.1016/j.dsm.2024.07.004.

[46] P. Dhunnoo, B. Kemp, K. McGuigan, B. Meskó, V. O'Rourke, and M. McCann, "Evaluation of telemedicine consultations using health outcomes and user attitudes and experiences: Scoping review," *J. Med. Internet Res.*, vol. 26, Jul. 2024, Art. no. e53266, doi: 10.2196/53266.

[47] S. Selvaraj and S. Sundaravaradhan, "Challenges and opportunities in IoT healthcare systems: A systematic review," *SN Appl. Sci.*, vol. 2, p. 139, Jan. 2020, doi: 10.1007/s42452-019-1925-y.

[48] P. Abril-Jiménez, B. Merino-Barbancho, G. Fico, J. C. M. Guirado, C. Vera-Muñoz, I. Mallo, I. Lombroni, M. F. C. Umpierrez, and M. T. A. Waldmeyer, "Evaluating IoT-based services to support patient empowerment in digital home hospitalization services," *Sensors*, vol. 23, no. 3, p. 1744, Feb. 2023, doi: 10.3390/s23031744.

[49] I. Rodríguez-Rodríguez, M. Campo-Valera, J.-V. Rodríguez, and A. Frisa-Rubio, "Constrained IoT-based machine learning for accurate glycemia forecasting in type 1 diabetes patients," *Sensors*, vol. 23, no. 7, p. 3665, Mar. 2023, doi: 10.3390/s23073665.

[50] K. Anderson, "The role of edge computing and hybrid clouds in next-generation healthcare," Univ. Cambridge, Cambridge, U.K., Tech. Rep., 2024. [Online]. Available: https://www.researchgate.net/publication/387220007_The_Role_of_Edge_Computing_and_Hybrid_Clouds_in_Next-Generation_Healthcare

[51] H. Kolb and S. Martin, "Environmental/lifestyle factors in the pathogenesis and prevention of type 2 diabetes," *BMC Med.*, vol. 15, no. 1, p. 131, Jul. 2017, doi: 10.1186/s12916-017-0901-x.

[52] H. Alamro, V. Bajic, M. T. Macvanin, E. R. Isenovic, T. Gojobori, M. Essack, and X. Gao, "Type 2 diabetes mellitus and its comorbidity, Alzheimer's disease: Identifying critical microRNA using machine learning," *Frontiers Endocrinol.*, vol. 13, pp. 1664–2392, Jan. 2023, doi: 10.3389/fendo.2022.1084656.

V. K. DALIYA received the M.Tech. degree in embedded systems technology. She is currently a part-time Research Scholar in the field of machine learning under the guidance of Dr. T. K. Ramesh with the Department of Electronics and Communication Engineering, Amrita School of Engineering, Bengaluru. She is also the Vertical Head of the Power Apps and Automate Division, CloudThat. She has more than 12 years of teaching and training experience in various technology domain and she has been managing a technical training team for the past two years. Her research works and guided projects are published in various IEEE conferences and in international journals. She has guided the academic project works of many M.Tech. and B.Tech. students. Her research interests include artificial intelligence, application of machine learning in healthcare, and data analysis in the IoT. She is a Life Member of the Indian Society for Technical Education.

T. K. RAMESH (Member, IEEE) received the Ph.D. degree in optical networks from Amrita Vishwa Vidyapeetham. He is currently a Professor with the Department of Electronics and Communication Engineering, Amrita School of Engineering, Bengaluru. He has 28 years of teaching and research experience. He has successfully guided eight Ph.D. students and is currently guiding eight Ph.D. scholars. He has published more than 100 research publications in peer-reviewed international journals and conferences. His research interests include communication networks and its applications, analog and digital devices and circuits, functional safety, artificial intelligence, and network-on-chip. He is a Lifetime Member of the Indian Society for Technical Education and a member of The Institution of Electronics and Telecommunication Engineers.