An Hybrid Ensemble Machine Learning Approach to Predict Type 2 Diabetes Mellitus

G. Geetha
Research Scholar, Sathyabama Institute of Science and Technology, India.
E-mail: geethag@srmist.edu.in

K. Mohana Prasad
Associate Professor, Sathyabama Institute of Science and Technology, India.
E-mail: mohan.cse@sathyabama.ac.in

Received November 13, 2020; Accepted December 27, 2020
ISSN: 1735-188X
DOI: 10.14704/WEB/V18SI02/WEB18074

Abstract

Diabetic Mellitus is one of the chronic diseases that affect many people around the globe. The severity of the disease and risk can be greatly reduced if it is predicted in the early stage. The main objective of the proposed model (T2DDP) is to predict type 2 diabetes mellitus and alert the patients well in advance to reduce the risk factor and severity associated with diabetes diseases. We have used supervised classification algorithms such as Naïve Bayes and ensemble algorithms like bagging with random forest and Adaboost for decision tree. The ensemble algorithm is mainly used to improve the performance by combining two or more models; this will aggregate the results of the entire model which can greatly enhance the accuracy and precision of predictions. Through this post, we are trying to build hybrid models that doctors can effectively use to treat diabetic patients. This method helps physicians to quickly group, identify and classify the disease type and manage it accordingly. Finally, the findings of the forecast will be submitted to the patient's cell phone at an early stage to make the immediate decisions about the health risk.

We also separated the data set into 1) training set and 2) evaluation set. The Pima Indian dataset was used to evaluate and interpret results, which involves n number of variables of medical predictors and one variable. Initially, our suggested approach is used to detect outer data, if applicable, using the Gaussian distribution method. After outlier detection, the missing values are filled out by taking the mean of the data rather than eliminating. We then split the dataset into different ratios of the training set and testing test to perform analysis on them: 85/15, 80/20, 70/30, 60/40. Naïve Bayes, bagging with random forest and Adaboost for decision tree is tested with a k10-fold cross-validation model for accuracy, precision, recall, and f1-score measures. Finally, we combined the predictions results of all the classifier models using stacking ensemble machine learning algorithms to increase the accuracy of the prediction.

Keywords

Diabetics, Machine Learning, Ensemble, Data Mining, Classification.

Introduction

The world iHealth organization (WHO) estimates that by i2019, more than i400 million people will have diabetes, and that number will exceed about i15 million. iMore than i2 million people have died due to its high blood glucose levels. The machine learning algorithm is the most efficient and promising approach for early prediction of any chronic disease. Type 2 diabetes mellitus (T2DM) is the most prevalent among 90 percentage of the diabetes patient and it leads to more complications, Basole(2015). The situations may go even worse, the youth and teenagers maybe get affected in the future. Hence it is necessary to develop a system to predict and treat the diabetes patients, CDCP(2017). Many diabetic disorders are triggered and damage our kidneys, skin, heart and other organs. We have to learn first how the body functions without diabetics to understand diabetics. The food we eat includes many components, including sucre, protein, fat, etc., and sugar is mainly extracted from carbohydrate products whose energy our bodies use. Breads, cereals, noodles, potatoes, fruit, dairy products and vegetables are high in carbohydrates. Such food is divided into glucose by our bodies as ingested and supplied in our bloodstream. Glucose is often used in the brain because, for the rest of our bodies like cells or the liver, this is needed for the body, A.D(2014). In order to function in a human body, insulin is an essential factor. It is a hormone formed in the pancreas by beta cells. It helps glucose to transfer to the cells of our body from the bloodstream. When insulin is used to make the pancreas, it requires ample glucose. If there isn't enough insulin in the pancreas, glucose is produced and diabetics are created in a human. Diabetics may have signs or symptoms of blurry vision, tirediness, weight loss, appetite and thirst changes, excessive urine, confusion, impaired recovery, frequent infections and concentration problems in the process.

Machine Learning Algorithm and its Types

There are three types of the machine learning algorithm, they are: 1) supervised learning, 2) unsupervised learning and 3) Reinforcement learning. Supervised learning is mapping an input variable(X) with an output variable(Y), it uses the labeled training data. Classification and regression fall under this category. Classification is used to predict the given samples when the output variable is in the form of categorical value whereas regression is used for real values. Ensembling also a supervised learning algorithm, this is used in case an individual algorithm does not provide more accurate results on a new sample. Random forest with bagging, boosting with XGboost are examples of this category.

The unsupervised learning algorithm is used when you have only an input variable(X) and there are no corresponding output variables, it uses the unlabelled training data. Association, clustering, and dimensionality reduction fall under this category.

The reinforcement learning algorithm learns by trial and error technique. Initially, it does not know what actions to be performed but over time it will learn by itself by trial and error, Aggarwal(2014). Ensemble machine learning algorithm provides more accurate results than other methods because it combines the output of two or more method for final prediction. Several studies have been conducted and it is proved to be a very suitable algorithm for predicting diabetes, Nai-arun (2015).

The obstacle for the learning field is outer data and imbalanced knowledge sets that limit the precision of the predictive model. By applying the isolation forest method (iForest) for the removal of the outlier data, Domingues(2018) and the Tomek relation (SMOTETomek) technology of synthetic minority over-sampling in order to stabilise the unbalanced data[9], the efficiency of the model was greatly enhanced.

We used controlled algorithms, such as Naïve Bayes, as well as assemble algorithms, such as random wood bagging and Adaboost for decision tree. A k-fold cross-validation model for accuracy, precision, retrieval and f1-score measures is available for Naïve Bayes, Randeom Forest, Bagging with random forest and Adaboost for decision tree. The bagging with random forest is considered to be well done in the prediction of the diabetic.

Diabetes disease prediction model (DDPM) is an early detection prediction model of diabetic type 2, based on human parameters. The DDPM system proposed is the most accurate, precise and reminder system. In order to draw precise conclusions, data analytics detect secret trends from vast volumes of data. Diverse machine learning algorithms are used in the area of healthcare to mainly screen input medical data and create models to predict. We are going to use this document to forecast diabetics with the aid of PIMAdataset using strategies such as naive Bayes and RandomForest algorithm.

Literature Review

This section is for understanding and identifying the research gaps in existing approaches by analyzing the existing literature survey. Why a machine learning algorithm is necessary for Big-Data analytics, especially in the Healthcare domain. What are the unresolved challenges in the healthcare sector, Vaishali(2017). The identifying risk factor is another dimension in the T2DM system. The Bayesian multi-task and feature relationship learning approach outperform the state of art model, Hasan(2020).

ALESSANDRO ALIBERTI et.al (2019). “A Multi-Patient Data-Driven Approach to Blood Glucose Prediction”. They have proposed to predict the blood glucose value of the new patients based on the training done on the existing patients. They have designed a non-linear autoregressive (NAR) neural network and on long short-term memory (LSTM) networks for blood glucose prediction.

PriyankaIndoria, YogeshKumar Rathore (2018). “A survey for detecting and predicting diabetics using machine learning techniques”, Priyanka(2018). In this paper, they proposed to use a machine learning algorithm like supervised, unsupervised, semi-supervised, reinforcement, and deep learning algorithms to enhance the accuracy of the prediction. And also they have compared the performance of the various algorithm.

Khaleel(2013) 'A data mining survey on the discovery of common diseases through medical data'. The emphasis of this paper is the retrieval of the medical knowledge to classify common diseases, such as cancer of the breast, heart disease, lung cancer, etc. Application was made of data mining techniques like Apriori and FPGrowth, linear genetic scheduling, algorithm of the decision tree, unattended neural networks.

K. R. Sasipriya, E. Deepa, Vembandasamy, E. (2015). "The goal is to test the use of the Naïve Bayesian algorithm for heart diseases". The algorithm used in this section is Naïve Bayes, which strongly believes that the existence of any attribute in a class is not connected to the presence and performance of any other attributes. WEKA is used and division into 70% of the percentage divide is achieved by splitting the results. 86.41% of the data entered in correctly and 13.58% of incorrect instances were collected using the ingenuous Bayese technique. He uses a knowledge collection of 500 instances or patients from a leading diabetes research institute in Chennai.

"An expert diabetes diagnostic system". TawfikSaeed Zeki(2012). They recommended an IF-THEN scheme based on the rules. Three components have been included, including the Block Diagrams, the Mockler Maps, and the Decision Tables. This system diagnoses diabetics by taking several things into consideration. InVP-Expert has been established.

In the "Fuzzy Expert Systems for Diabetic Diagnosis Comparative Analysis," Vishali Bhandari and Rajeev Kumar(2015) compared various fuzzy systems of experts by using several diabetic criteria. For the comparative analyses of these expert systems, the MATLAB fuzzy logic toolbox was used. Five comparison parameters were used and results produced.

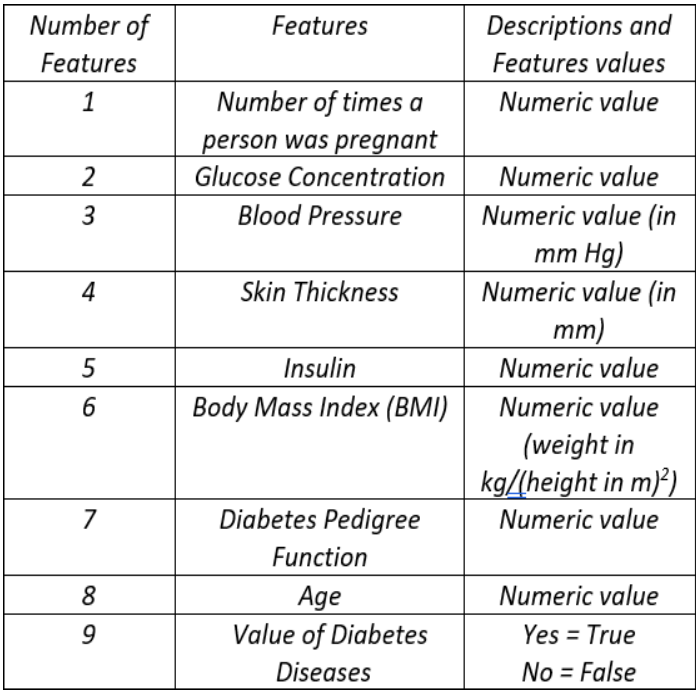
The "Machine Learning and Data Mining Methods of Diabetic Research," by Ioannis Kavakiotis and Olga Tsave(2017). Data mining techniques and algorithms were extensively researched into the estimation and detection, complication, and health treatment of general data features of diabetics. Data mining techniques. The methods part of this approach are referred to as filter methods and before model development, the feature set is filtered out.

Amelga, Marco M. Maribondang, MulyadiSalim(2016), "Cardinal Risk Detection for adult diseases using naive Bayes Classifier". The detection of risk of cardiovascular diseases is not a choice. The aim of this paper is to classify the extent of risk of cardiovascular disease with the Naïve Bayes classification. Any of the characteristics of cardiovascular disease from this risk level may be determined as the key risk factors such as diabetic Mellitus, coronary-artery function, renal function and the level of lipids in the blood. According to the principles of these risk factors, class codes should be assigned: risk level 1, risk level 2 and so on. The assessment of this approach was performed with precision, sensitivity and specificity in three parameters. 80% of the findings were correct in the model proposed. A variety of machine learning approaches, such as naïve bays, decision trees, sorting or clustering, and neural networks have been performed. The findings revealed that the highest accuracy rating is found in all the naïve Bays.

"The Machine Learning Framework to Electernal Health Registers for Type 2 Diabetics," ZhengT, XieW, Xu L, He X, Zhang Y, You M, Yang G, Chen Y(2017). This paper attempts to classify diabetics of type 2 (T2DM) via electronic health records (ERH). Different interactions between genotypes associated with type 2 diabetics can be established through means of phenome-wide interaction (PheWAS) analysis and GWAS tests and controls (for example, via an Electronic Health Records). They created a semi-automated architecture with a machine learning algorithm to increase the retrieval rate, holding the wrong rate low. They suggested a system that specified the problems of engineering and machine learning with or without ERH's T2DM. In order to calculate individual success metrics the following learning models are measured and contrasted: Random Forestry, Naïve Bayes, Logistical Regression, K-Nearest- Neighbor and Help Vector Machine. They also used a collection of 300 reports of random EHR archives from 23,281 patients with diabetics.

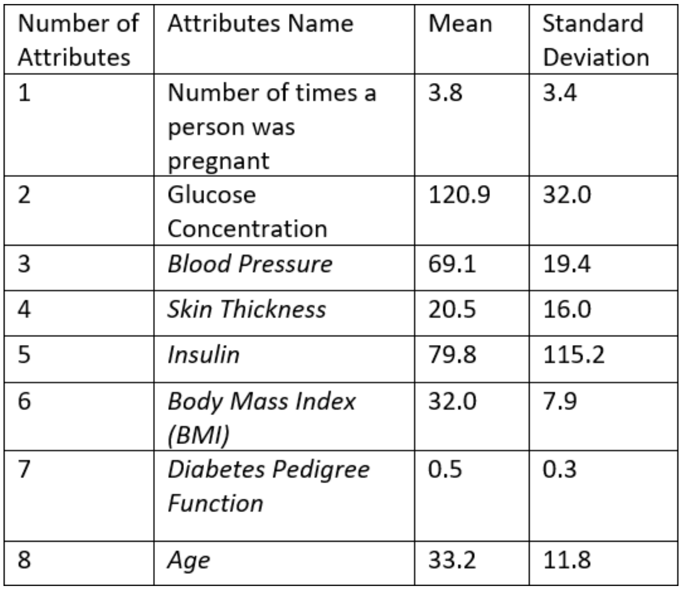
"Mellitus diabetics affected Patients Classification and Machine Learning Techniques Diagnose" Francesco Mercaldo. VittoriaNardone, AntonellaSantone (2016). The aim of this paper is to distinguish between diabetic people from diabetic people. They also carried out hypotheses using multiple characteristics of diabetic and diabetic citizens. The methods used were Mann-Whitney (with p-level adhered to 0.05) and Kolmogorov-Smirnov, to search for null hypotheses (With p-level adhered to 0.05). They selected a significance level of 0.05 and six algorithms of classification, namely Hoeffding Tree, Random Forest, JRip, Multilayer Perceptron, and Bayes Network. The effects are calculated by accuracy, recall estimation, F-measurement and ROC range.

Proposed Methodology



Number of Features	*Features*	*Descriptions and Features values*
1	*Number of times a person was pregnant*	*Numeric value*
2	*Glucose Concentration*	*Numeric value*
3	*Blood Pressure*	*Numeric value (in mm Hg)*
4	*Skin Thickness*	*Numeric value (in mm)*
5	*Insulin*	*Numeric value*
6	*Body Mass Index (BMI)*	*Numeric value (weight in kg/(height in m)^{2})*
7	*Diabetes Pedigree Function*	*Numeric value*
8	*Age*	*Numeric value*
9	*Value of Diabetes Diseases*	*Yes = True* *No = False*

Fig. 1 Features of Pima Indians dataset for Diagnosing Type 2 Diabetics Disease



Number of Attributes	Attributes Name	Mean	Standard Deviation
1	Number of times a person was pregnant	3.8	3.4
2	Glucose Concentration	120.9	32.0
3	*Blood Pressure*	69.1	19.4
4	*Skin Thickness*	20.5	16.0
5	*Insulin*	79.8	115.2
6	*Body Mass Index (BMI)*	32.0	7.9
7	*Diabetes Pedigree Function*	0.5	0.3
8	*Age*	33.2	11.8

Fig. 2 Dataset variables and their values

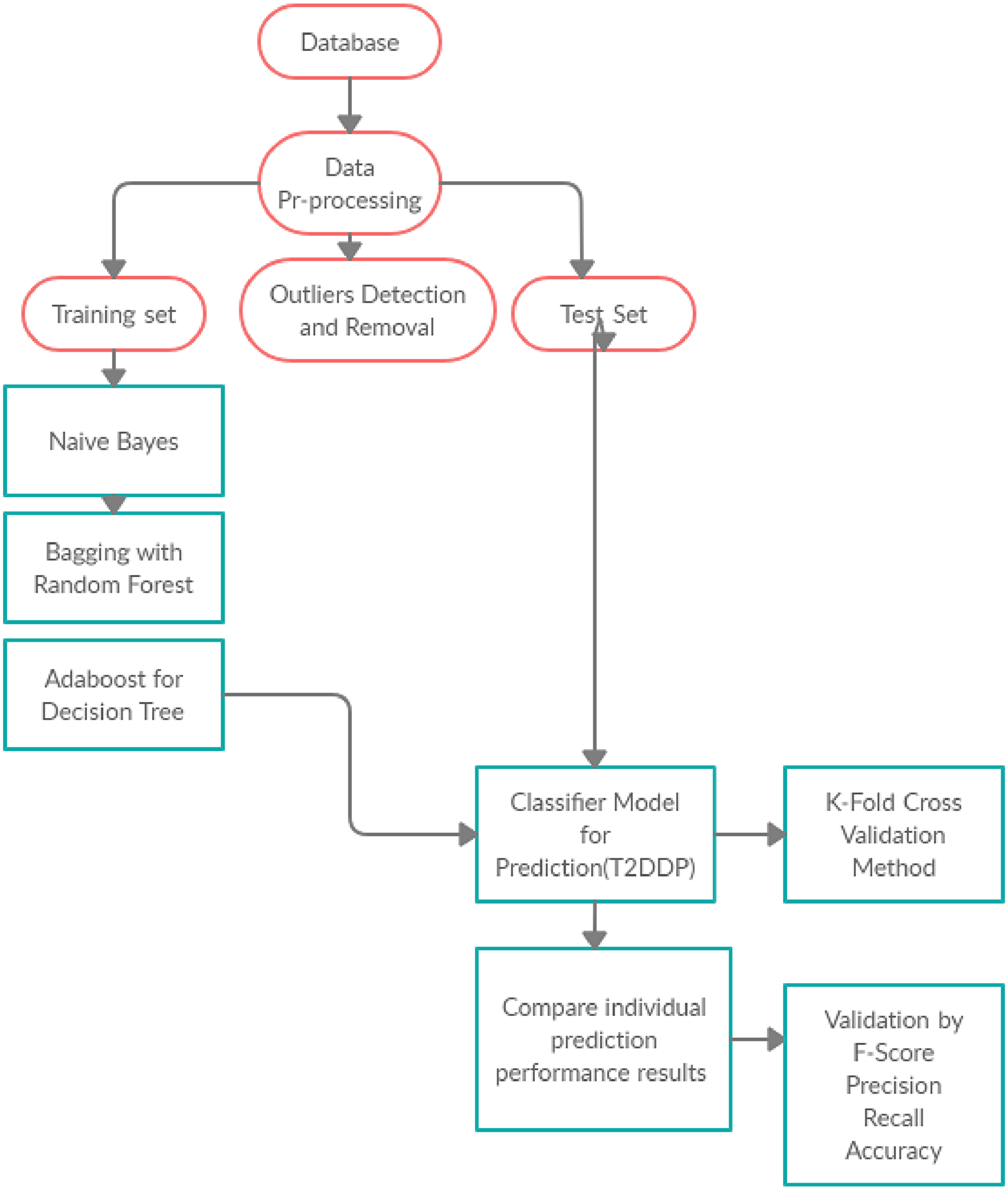


Fig. 3 Process Flow Diagram of Proposed Methodology

Algorithm of Proposed Methodology

Step 1: Select the dataset and pre-process for outlier detection and removal

Step 2: Divide the dataset into k-fold, where k=10

Step 3: Select the supervised and ensemble learning algorithm, i.e, Naïve Bayes, Bagging with Random Forest, Adaboost for Decision Tree
Step 4: Train the classifier model by keeping one fold as test data and the remaining fold as training data and test the trained model with the fold reserved as a test set.
Step 5: Repeat step 4 for the ‘k’ cycle
Step 6: Evaluate the results obtained from all the models using the accuracy, precision, recall, and f-score measures.
Step 7: Combine the predictions results of all the classifier models using stacking ensemble machine learning algorithms to increase the accuracy of the prediction.

Procuring the Dataset

The PIMA Indian Dataset is the dataset used here. That is the data gathered by the State Diabetics Institute. It consists of multiple variables and one target variable for medical predictors. Different physiological variables: BMI, blood pressure, glucose levels.etc. There are 768 columns and 9 sides. The file is open with.csv(Comma Segregated Values). We import the file into our Python environment using the Pandas embedded library, a data structure library. The remaining libraries imported into the environment are:

Numpy – a library which mainly uses large-size arrays and matrices to operate on data with high-level mathematical functionality.
Matplotlib – a library that offers plotting diagrams and plots for Python. It functions for NumPy in combination. Pandas are called read csv(), which simply reads a format file (.csv).

When the dataset is loaded into the environment, the.shape() function that returns the row and column number helps one to verify the dimensions of the dataset. The simple data search is carried out using the built-in.head() and.tail() that prints the number of rows from the beginning of a data set and from the bottom.

Preparation of the Dataset

We will see how we can make some adjustments to the dataset after receiving the dataset. Activities including initialization of the variables, rejection of outliers, cleaning of data, collection of features and the planning of suitable data marking take place. The dataset includes a skin thickness parameter in our situation, which has a poor association of a person's diabetic contribution. So for our review, we delete the column. At this point, the numerical dimensions of the details, for example, the average of a column, the numbers of

cases of columns on the basis of conditions, etc., can be determined. Therefore, for each case, we determined the number and it turned out that:
People with Diabetics: 268 People without Diabetics: 500

During data pre-processing, the irrelevant data has been removed first then, outliers such as unwanted data like out of range data are removed using Gaussian distribution. In Gaussian distribution, using modified z-scores, errors are detected and removed. The standardization is done by measuring z-score

$$\mathrm{Z}_{\mathrm{i}}=\frac{0.6745(xi-x')}{MAD}$$

where MAD is the median absolute deviation, ‘x’ denotes the median.
After outlier detection, the missing values are filled out by taking the mean of the data rather than eliminating it by using the following equations [22]:

$$\mathrm{Q(X)}=\begin{cases} maean\{x\}\ if\ x = null/missed \\ \qquad\qquad x\ otherwise \end{cases}$$

The dataset contains values for those with and without diabetes. In the given data, around 35% of the people have been diagnosed with diabetics.

Splitting the Data

One of the key steps in the study is to split the data collection into training and test data. This method means that the evaluation results are distinct from the testing data so the model adopted by the formation process has to be checked. Second, the data of preparation is studied and then the data that is taught is generalised to the other data from which the forecast is made. The data set is separated into many versions in our case and a forecast is carried out accordingly. The dataset has numerous columns of the outcomes of diabetics which are medical predictors and a goal column. Health predictors are presented as inputs for a variable and the target variable as feedback to a related variable.

The data set is grouped into arrays and is matched to training and test subcategories using the built-in feature train test split. In our case, we do 80/20,70/30,75/25,60/40 splits and each of these is reported in their precision. It was found that the dataset has certain null values, and the null values have been filled with the mean value Q(X) of the respective columns to streamline the analyses and forecasts.

Naïve Bayes

The Baïve Bayes Network is the most commonly encountered and shortest type of classification network. The Baïve Bayes theorem holds, assuming strongly that a class has some particular characteristic regardless of any other feature. The classification requires very little preparation and is well carried out in categorical data. But if any variable not observed during the training time will be considered as zero probabilities by the model, this zero-frequency problem can be rectified by the smoothing technique like Laplace estimation.

The Naïve Bayesian is based on the conditional probability (given a set of features, the probability of occurrence of certain results):
Given a set of features, X = {x1,x2,x3...,xd}, the posterior probability for the possible outcomes Cj could be constructed based on the conceivable results C = {c1,c2,c...,cd}, where X is the predictor variable and C is the arrangement of absolute dimensions of the needy variable. Utilizing Bayes' standard:

$$P(C_j \mid x_1, x_2, \ldots, x_d) \propto P(x_1, x_2, \ldots, x_d \mid C_j)\, p(C_j)$$

where p(Cj | x1,x2,x..., xd) is the posterior probability of class i.e., the probability that X has a place with Cj. Since Naive Bayes does not perform well for large samples that is if there are more number of features and if it takes more values, to overcome this using Bayes theorem the above conditional probabilities could be decomposed into:

$$p(X \mid Cj) \propto \Pi\, p(xk \mid Cj)$$

and revised the posterior probabilities is:

$$p(\ Cj \mid X) \propto p(c_j)\, \Pi\, p(xk \mid Cj)$$

The naïve Bayes algorithm first constructs a frequency table from the dataset and then creates the likelihood table by finding the probabilities of the occurrences. Then find out the posterior probabilities for each class using the Bayesian equation. Lastly, the class with the highest probabilities will be the outcome of the prediction. This estimates the consistency of the model by comparing the model proposal with the original. The metrics library in sklearn is used for this purpose.

Bagging with Random Forest

The Algorithm Random Forests is the most powerful grading algorithm and can grade a huge number of results with the most precision. Random Forest is a Community learning instrument (this is some kind of neighbouring predictor) for classification and regression which creates various decision makers during training and then is referred to as the predictor.

In this technique, firstly we have to create the multiple models using the Bootstrap sampling method on the data sets. That is a Bootstrap sampling method constructs multiple training sets from the original datasets. The size of the training sets N is equal to the size of the original datasets M. The training set is the random subsamples of the original dataset, which may contain the same records multiple times, or some records are not all added. The original dataset will be used as the test dataset.

The variables for tree predictors for each tree will depend on the values of a random vector sampled separately for all trees of the forest with the same distribution. Random Forests addressed the issue by seeking a normal equilibrium between the two extremes, which is high variance and high partiality. There is also a mechanism for estimating error rates (Out of the Bag error).

Out of Bag error is a measure of prediction error which will be calculated by averaging the prediction error of all unused samples from each bag.

Many machinery learning models are readily influenced by the outliers of the training data such as linear and logistic regression. Outliers are variations in the behaviour of the device which may often be triggered by mistakes of the human instrument. It is possible to contaminate a single sample. These outlines or extreme values do not influence the performance/precision of the model. This dilemma is tackled by the RF algorithm and resolved, Karthick(2020).

In our case, the mark is separated into two variables, and this is the classification input. The ability to use any data in particular for the collection of features is one of the most significant strengths of Random Forest classificators.

In our case, we in the sklearn library use the RandomForestClassifier() function to predict the data. The input and test data are fitted to the model (). During estimation, training data is then split into arrays. Through comparing the expected values with the initial sets of values, the exactness of the formula is achieved.

Adaboost for Decision Tree

The major difference between AdaBoost and Bagging is, bagging construct trees in all the iterations and finally, the tree which receives higher votes will be the considered best-performed tree whereas Adaboost built tree based on past misclassification error. Every model learns error or mistakes done in the previous model and rectified it while it constructs the next model. Bagging constructs tree in a parallel manner whereas Adaboost built tree in the sequential manner Decision tree is the most interesting and very popular machine learning algorithm which works similar to the human brain. Adaboost for decision tree algorithm takes N samples from the dataset, target variable Y=1,-1, and R covariates. Assign weights for each row as 1/N and trained the classifier to find out the misclassified error.

$$E= \sum_i^N Wi * \pi(yi \neq G1(xi)/\sum_i^N Wi$$

In the above equation, we have self-normalized the weighted error and it falls in the range 0 to 1. From this 'importance of say' is calculated by the equation $\alpha=\log\frac{(1-E)}{E}$. If this procedure repeats for n iterations, in the last iterations we got n classifier with the weighted vote. To predict for new observations, the equation is y=sign ($\sum_j^m \alpha G(x)$).

T2DDP Model

We have compared the prediction accuracy of the entire classifier models which were discussed above and concluded that our proposed T2DDP model achieved high accuracy. The individual model has produced less accurate results but it is improved by combining the outputs of all the classifiers based on the prediction accuracy using the "stacking ensemble machine algorithm". The architecture of stacking has two levels, the first level uses two or more base models called the level 0 model or base model, and the second level called a meta-model or level 1 model combines the two or more predictions of the base model. The Metamodel is trained without sample data. The common approach used to construct the training dataset for the Meta model is k-fold cross-validation. After building the training dataset, Metamodel will be trained on this dataset whereas the base models are trained in the original dataset. In the base model, the following classifiers are used Naïve Bayes, Random forest, and Adaboost with a decision tree. Next, meta model combines the outputs all the classifier models to produce the fnal results.

Cross-Validation Model

Underfitting and Overfitting: Overfitting occurs when the model fits the data exactly but it will not perform so well when a new dataset is used but whereas underfitting occurs when the model does not fit the data correctly. Underfitting often arises due to poor pre-processing. Overfitting can be eliminated by cross-validation methods like k-fold cross-validation and stratified k-fold cross-validation method etc.

To validate the capability of the model, there are two validation methods, namely the hold-out method and the k-fold method. In a hold-out method, it divides the dataset into two parts, first one is the training set which is used for training the machine learning algorithm and the second one is the testing dataset which is used for evaluating the model. But this method is not reliable, hence used the k-fold cross-validation technique to evaluate our model. In k-fold cross-validation, it divides the samples into k subsamples, for example, if k=5 then the dataset is divided into 5 folds. If k fold is used as a test set then k-1 folds are used as a training set, likewise, all the folds are used as the test set and training set. That means all the data in the datasets are tested as well as trained which increases the accuracy and reduces the higher variance. The performance metric was estimated by the following equation, Amith Sai(2020)

$$\mathrm{V(X)}=\frac{1}{K}\ \mathrm{x}\ \sum_{n=1}^{k} Rn \pm \sqrt{\sum_{n=1}^{k}\frac{(Rn-R)2}{K-1}}$$

Finding the Accuracy

First, by presenting argument for the break of training data, the accuracy of training data is verified. The detailed test results was then conducted with the test data in the same manner as the parameters. We may create an uncertainty matrix by comparing these two. The key purpose of the uncertainty matrix is to determine the classification's accuracy. The uncertainty matrix C is, by implication, the Ci,j is the number of observations known in the group I but expected in the group j. Thus the count of true negatives in binary classification is C0,0; false negatives are C1,0 and true positives are C1,1 and false positives are C0,1.

Metrics used to Evaluate the Implementation Method

To evaluate the implementation model the following parameter were used: precision, recall, accuracy, and f1 score are used. These parameters are evaluated based on true positive (TP), true negative (TN), false positive(FP), and false-negative(FN).

$$\text{Accuracy}=\frac{TP+TN}{\substack{(TP\\ TP+TN+FP+FN)}}$$

$$\text{Precision}=\frac{TP}{(TP+FP)}$$

$$\text{Recall}=\frac{TP}{(GP+FN)}$$

$$\text{F1-score}=2.\frac{precision.recall}{precision+recall}$$

Experiments and Results

A new policy-based entry system into the Fog computing area has been implemented to provide the differentiated service to the IOT implementation consumer. Instead of bringing the customer to the cloud setting, the customer can get a custom service from Fog to run their delicate data quicker on its edge network. This service enhances the Internet service provider and cloud service provider's capabilities to the top of the network. This service extension allows for the vibrant resource provisioning based on policies through the edge network virtualization function. The virtualization idea will contribute to the creation of a versatile and managed network setting based on policy as shown in Fig.4, by implementing new techniques such as "Software Defined Network (SDN)", Ponnusamy(2020) and "Network function virtualization (NFV)", Nandakumar(2020) In Figure 4, the integration of the policies scheme into the IoT scheme is shown. In Figure 4, the policy repository that has all the policies and norms for the management of an IoT system is an Master Policy server. The policy cabinet with customer interface for store measures and regulations like routing, QoS, resource distribution and decision-making in relation to the policy repository. The Policy Decision Point (PDP) module, which provies information to Master Policy Server(MPS), implements all policy decisions.

The policy choices are transferred to the policy manager of Fog. Second, the domain policy server has been provided for Fog routers. The Fog domain server had the Fog domain manager which had MPS manager policies and guidelines for routing, QoS, security and smart decision making. The Fog domain policy server with these Fog domain executives have been given the Policy Enforcement Point to implement Fog's policy choice for decision-making and intervention. Moreover, Fog Domain Manager had the power to transfer domain management measures below its implementation hierarchy. In addition, before being sent for policy processing and action evaluation, the Fog Manager would ask the Fog routers for the accessibility of resources. The Fog domain policy manager was informed about any anomaly in the Fog router. Thirdly, the Domain policies server is available to the wireless mesh routers. The domain policy managers were delegated by the Fog Manager to the domain manager who has a rule implementing the policy and guidelines on QoS, routing and security for wireless routers.

In Figure 4, R1, R2, R3….. Rn representes the local router within the local network which form the cluster. And all these routers are connected to Cluster Head (CH). These cluster head are responsible for transferring data to higher level for either processing or for storing. Finally, the domain policy server would be informed of any anomalies in the wireless router. Policies that are delegated to the applications below the structure of the corresponding executives in the policy oriented design for the IoT scheme and assign strategies to a realm below its structure.

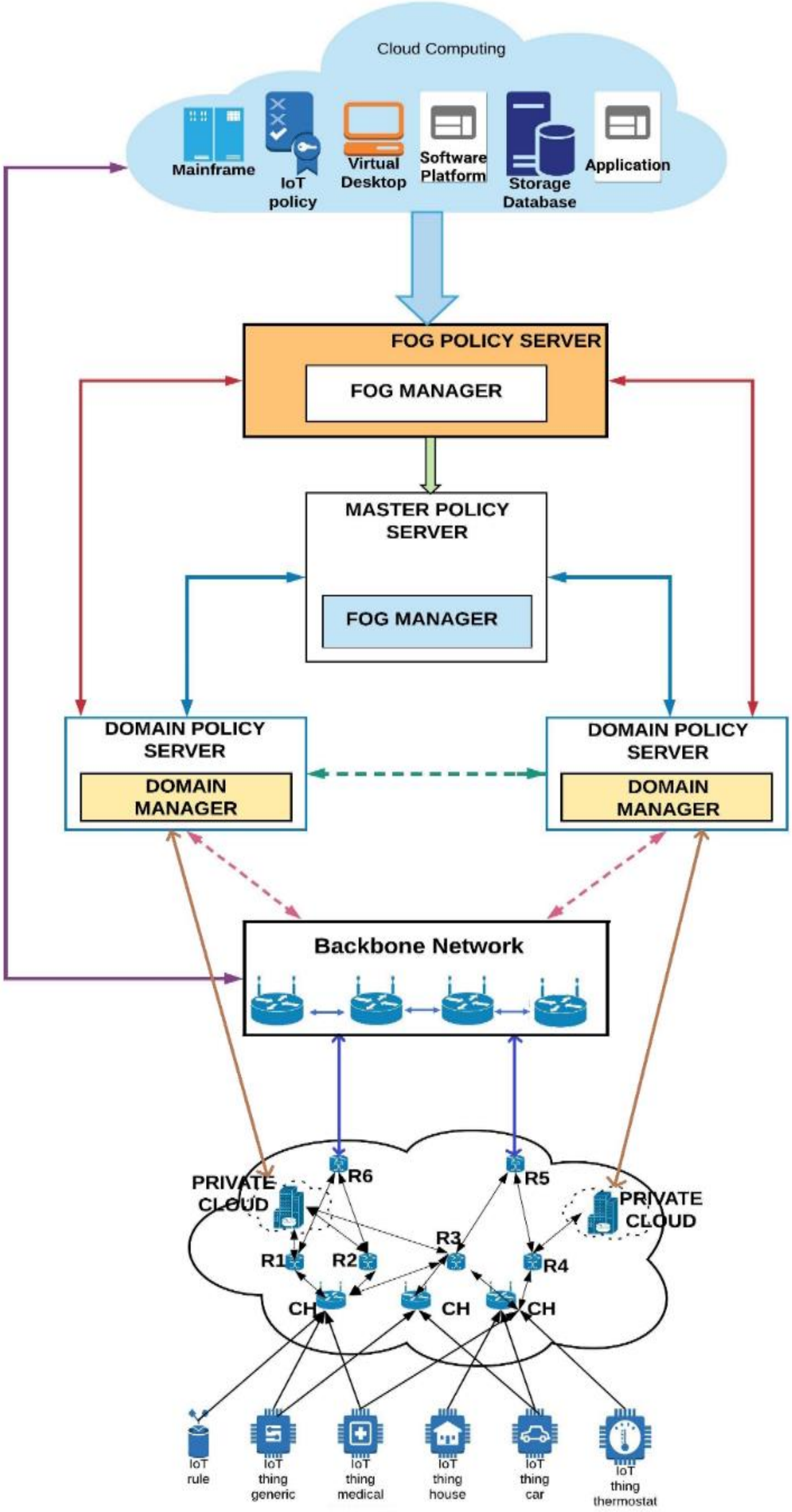


Figure 4 Fog Assisted IoT Based Medical Cyber System

We produce the following results for separate splits of training and testing data after running the Random Forest and Naive Bayes algorithms:

Figure 6 reveals that we get nearly 75% in the training sets and 74-77% in the test results on all four separations.

This indicates that the testing collection is trained to be up to 75% accurate, which means that the data trained have been used to anticipate test outcomes with an overall precision of 75% in the study.

In figure 7, we can see that for the four different splits, we get results that are close to 98% in the training set and 92-94% in the test results.

Table 1 Comparison of various splits using Naïve Bayes

Train/Test Split	Train output	Test output
60/40	75.22	77.27
70/30	75.98	74.89
75/25	75.87	74.48
80/20	75.57	77.27

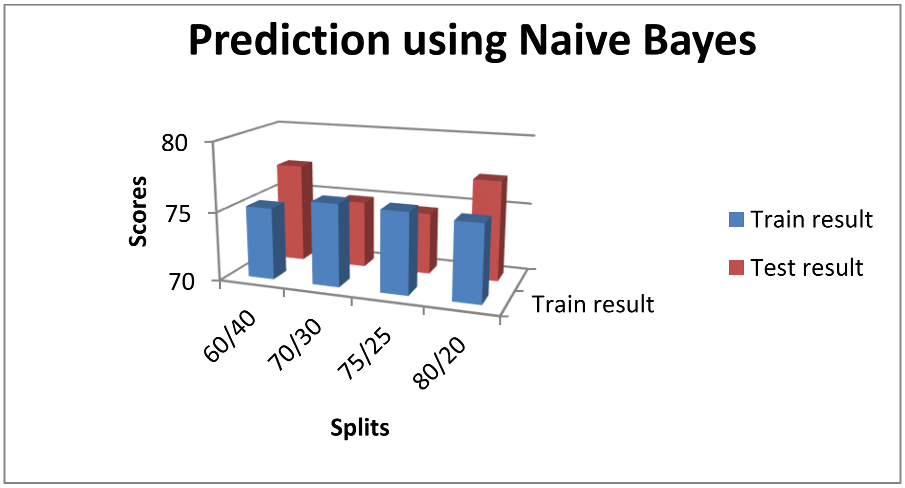


Fig. 5 Prediction using Naive Bayes

Table 2 Comparison of various splits using Random Forest

Train/Test Splits	Train output	Test output
60/40	97.61	92.27
70/30	98.7	92.16
75/25	98.61	93.92
80/20	98.4	94.37

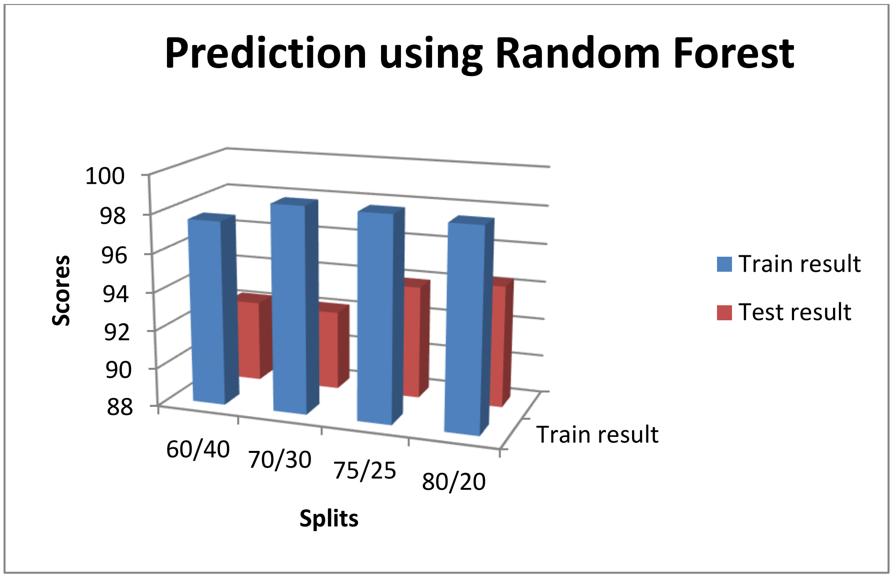


Fig. 6 Prediction using Random Forest

Table 3 Comparison of various splits using AdaBoost with Decision Tree

Train/Test	**Train result**	**Test result**
60/40	94.61	91.27
70/30	95.7	92.96
75/25	96.61	92.92
80/20	97.37	94.37

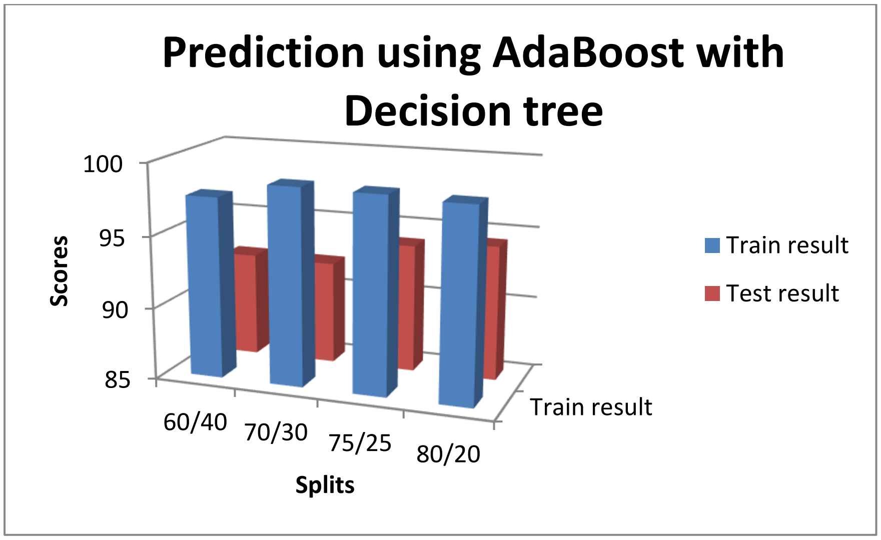


Fig. 7 Prediction using Adaboost with Decision Tree

Table 4 Comparison of various splits using T2DDP

Train/Test	**Train result**	**Test result**
60/40	97.61	96.27
70/30	97.7	95.96
75/25	98.61	97.92
80/20	98.37	98.37

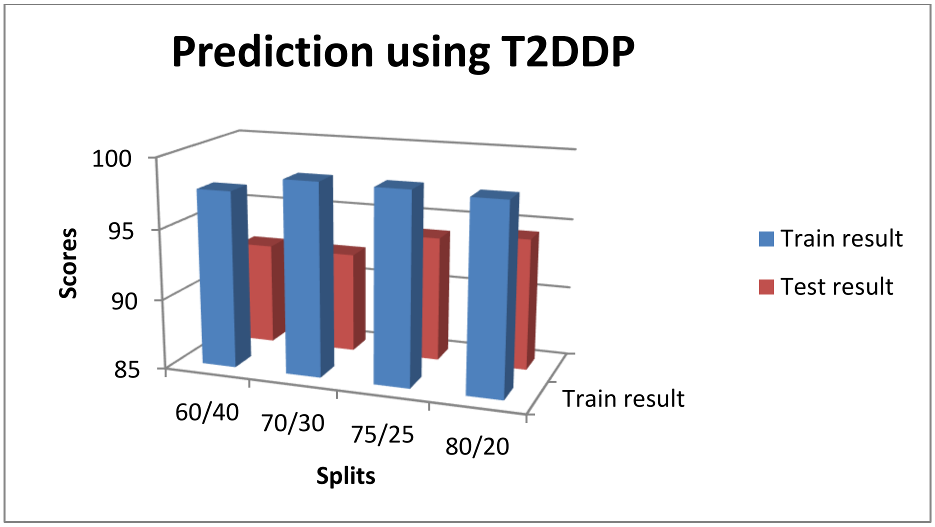


Fig. 8 Prediction using T2DDP

In evaluating the outputs of all classification models, we can explain the better effects of training given by the proposed T2DDP model, which in turn increases analysis and prediction precision. The data collection is educated to the utmost precision, under which the Random Forest algorithm guarantees that no data are lost in large datasets without removing the data missing. The algorithm of Naïve Bayes seems not to know of incomplete data that does not provide detailed results in the study. The comparison diagram for the training outcomes of all the classification model as seen in Figure 8, covering Naïve Bayes and Random Woodland, Adaboost, the decision tree for separate splits. Compared to the other classification models, it can be shown that the T2DDP testing findings are reliable, as 98 percent of data is accurate when it comes to training.

After evaluating the results, the T2DDP is a more effective approach for analysing the dataset using the tools to segment it into training and testsets. It is a more detailed way to predict diabetics.

The results of Naïve Bayes' testing in the decision tree are exceptionally poor relative to the Random Forest and Adaboost, as errors exist in the training algorithm of Naïve Bayes. It can also find missing data and then the accuracy is fluctuation and error, but it provides accurate accuracy even when it comes to large data sets such as the PIMA dataset for Random Forest and Adaboost.

Table 5 Performance of Classification model

Methods	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Naïve Bayes	75.11	74.09	76.18	75.06
Bagging with Random Forest	93.08	92.23	94.23	92.78
Adaboost for Decision Tree	92.16	91.07	93.45	91.98
Proposed T2DDP model	96.56	96.12	97.58	95.98

Conclusion

Identifying diabetics or predicting the upcoming of a diabetic life can be propelled by using various machine learning techniques like Naïve Bayesian Network, Random Forest, and Adaboost for decision treeetc. In this paper, three machine learning algorithms such as Naïve Bayes, Bagging with Random Forest, and Adaboost for Decision Tree has been implemented for predicting diabetes. All these classifier models are evaluated by using the following measures: Accuracy, precision, recall, and F-score. We conclude that T2DDP is the best way to diagnose diabetes. This method gives us approximate results after splitting and analyzing training and test data. The efficiency of this method is much better than all other classifier model.

The PIMADataset research attempts to separate the dataset by finding the highest/best precision of the algorithms and how to respond if the data division is different. The dataset is collected in order to ensure the data set does not contain null values such that the prediction model is correct. Dataset preprocessing guarantees that all attributes (columns) are considered in the predicting process. From the forecast and study above, one can observe that the results obtained with the T2DDP are 98 percent correct. This approach is also an optimal way to analyse and forecast diabetics. This method is proposed.

Future Scope

Owing to the lack of equipment and services, medical professions found it impossible to find and interpret healthcare records. However, we can solve this with ML and carry out real-time data analysis that results in better modelling and predictions. It strengthens and increases health care in general. Now IoTs are combined with ML to make intelligent health care devices that make it easier for the user to adjust his body, health details by using the system (Pacemaker, Stethoscope, etc.). This helps to quickly track, predict and evaluate mistakes, saving people time and life.

References

World Health Organization, (2016). *Global report on diabetes*.

International Diabetes Federation, (2017). *Idf diabetes atlas*

Basole, RC., Braunstein, M.L., & Sun, J. (2015). Data and analytics challenges for a learning healthcare system. *Journal of Data and Information Quality (JDIQ)*, *6*(2-3), 1-4.

Centers for Disease Control and Prevention. (2017). National diabetes statistics report: Estimates of diabetes and its burden in the united states, *National diabetes statistics report*.

Association, A.D. (2010). *Diagnosis and classification of diabetes mellitus Diabetes Care*, *33*(1), 62-69.

Aggarwal, C.C. (2014). Ed., *Data Classification: Algorithms and Applications*. Boca Raton, FL, USA: CRC Press.

Nai-arun, N., & Moungmai, R. (2015). Comparison of classifiers for the risk of diabetes prediction. *Procedia Computer Science*, *69*, 132-142.

Domingues, R., Filippone, M., Michiardi, P., & Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition*, *74*, 406-421.

Goel, G., Maguire, L., Li, Y., & McLoone, S. (2013). Evaluation of sampling methods for learning from imbalanced data. *In International Conference on Intelligent Computing*, 392-401.

Vaishali, R., Sasikala, R., Ramasubbareddy, S., Remya, S., & Nalluri, S. (2017). Genetic algorithm based feature selection and MOE Fuzzy classification algorithm on Pima Indians Diabetes dataset. *In IEEE international conference on computing networking and informatics (ICCNI)*, 1-5.

Hasan, M.K., Alam, M.A., Das, D., Hossain, E., & Hasan, M. (2020). Diabetes prediction using ensembling of different machine learning classifiers. *IEEE Access*, *8*, 76516-76531.

Aliberti, A., Pupillo, I., Terna, S., Macii, E., Di Cataldo, S., Patti, E., & Acquaviva, A. (2019). A multi-patient data-driven approach to blood glucose prediction. *IEEE Access*, *7*, 69311-69325.

Priyanka, I., & Yogesh Kumar, R. (2018). A survey: detection and prediction of diabetes using machine learning techniques. *International Journal of Engineering Research & Technology (IJERT)*, *7*(3), 287-291.

Khaleel, M.A., Pradhan, S.K., Dash. G.N. (2013). A Survey of Data Mining Techniques on Medical Data for finding frequent diseases. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(8), 149-153.

Vembandasamy, K., Sasipriya, R., & Deepa, E. (2015). Heart diseases detection using Naive Bayes algorithm. *International Journal of Innovative Science, Engineering & Technology*, *2*(9), 441-444.

Tawfik, S.Z., Mohammad, V.M., Yousef, A.S., & Talayeh, T. (2012). An Expert System for Diabetics Diagnosis. *American Academic & Scholarly Research Journal*, *4*(5), 1-13.

Bhandari, V., & Kumar, R. (2015). Comparative analysis of fuzzy expert systems for diabetic diagnosis. *International Journal of Computer Applications*, *132*(6), 8-14.

Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., & Chouvarda, I. (2017). Machine learning and data mining methods in diabetes research. *Computational and structural biotechnology journal*, *15*, 104-116.

Miranda, E., Irwansyah, E., Amelga, A.Y., Maribondang, M.M., & Salim, M. (2016). Detection of cardiovascular disease risk's level for adults using naive Bayes classifier. *Healthcare informatics research*, *22*(3), 196.

Zheng, T., Xie, W., Xu, L., He, X., Zhang, Y., You, M., & Chen, Y. (2017). A machine learning-based framework to identify type 2 diabetes through electronic health records. *International journal of medical informatics*, *97*, 120-127.

Mercaldo, F., Nardone, V., & Santone, A. (2017). Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. *Procedia computer* *science*, *112*, 2519-2528.

Anandan, M., Manikandan, M., & Karthick, T. (2020). Advanced Indoor and Outdoor Navigation System for Blind People Using Raspberry-Pi. *Journal of Internet Technology*, *21*(1), 183-195.

Karthick, T., Amith Sai, A.V., Kavitha, P., Jothicharan, J., Kirthiga Devi, T. (2020). Emotion detection and therapy system using chatbot. *International journal of Advanced Trends in Computer Science and Engineering*, *9*(4), 5973-5978.

Ponnusamy, V., Kottursamy, K., Karthick, T., Mukeshkrishnan, M.B., Malathi, D., & Ahanger, T.A. (2020). Primary user emulation attack mitigation using neural network. *Computers & Electrical Engineering*, *88*, 106849.

Karthick, T., Nandakumar, R., Yuvaraj., & Ponnusamy, V. (2020). Data-driven methods for next generation of wireless communication networks, *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(4), 4696–4700.