International Journal of Recent Research and Review, Vol. XVI, Issue 1, March 2023

ISSN 2277 – 8322

An in-depth comparison of classification algorithms for diabetes prediction

Abishek Bhatta
Kathmandu, Nepal
abishekbhatta003@gmail.com

***Abstract*—Diabetes is a major health issue today, becoming a leading cause of cardiovascular and kidney-related diseases. Each year, 1.5 million deaths around the world are directly attributed to diabetes. Thus, there is a strong need for an early diagnosis of the disease so that people affected by it can take appropriate measures without delay and improve their overall health conditions. In order to facilitate early diagnosis, one solution is to implement classification algorithms. However, a critical challenge lies in the accuracy of these algorithms, i.e., they have varying performance and success rates depending on the nature of the dataset used to train the model. Therefore, to figure out the most efficient algorithm, this study tests classifiers like K-nearest neighbors, Naïve Bayes, Support Vector Machine, and XGBoost on a Pima Indian Diabetes Dataset (PIDD). The dataset was obtained from the United States National Institute of Diabetes and Digestive and Kidney Diseases. After testing each classifier, the study found XGBoost to be the most effective machine learning model for predicting diabetes.**

***Keywords*—Diabetes, Machine learning, Classification algorithm, K-nearest neighbors, Naïve Bayes, Support Vector Machine,XGBoost Classifier**

I. INTRODUCTION

Diabetes is a non-communicable disease caused by insufficient production of insulin in our body. It can lead to various long-term complications such as heart attack, stroke, nerve damage, and kidney failure [1]. In 2021, around 537 million people were found living with diabetes worldwide, with the number expected to rise to 643 million by 2030 [2]. Similarly, according to the WHO, 1.5 million deaths are directly attributed to diabetes each year [3]. These statistics imply that diabetes today has become a serious health problem, requiring early attention and care.

However, the correct diagnosis of diabetes at the initial stage can be challenging to doctors because, during manual decision-making, the hidden pattern of data can go unnoticed, which can impact the accuracy of the decision [4]. Therefore, there is a strong need for automated detection of diabetes with better accuracy, and to solve that problem, the classification method can be implemented. This machine learning technique uses the trained data to make predictions. First, a dataset is split into features and targets. Features are the input data or attributes of the dataset, whereas targets are the output data or categorical values. Once the dataset is split, both input and output data are passed to train the models using a particular classification algorithm. After the model is fully trained, it then predicts the categorical value of the given input data.

There are various classification algorithms today that can be used for predicting diabetes, but each of them has a different success rate, which is heavily influenced by the dataset used [5]. Hence, each algorithm must be tested against the diabetes dataset in order to evaluate their performance and identify the most suitable algorithm that can correctly predict diabetes. In this study, however, fourclassification algorithms (K-nearest neighbors, Naïve Bayes, Support Vector Machine, and XGBoost) are tested and evaluated on the basis of accuracy, precision, recall, and F1 score.

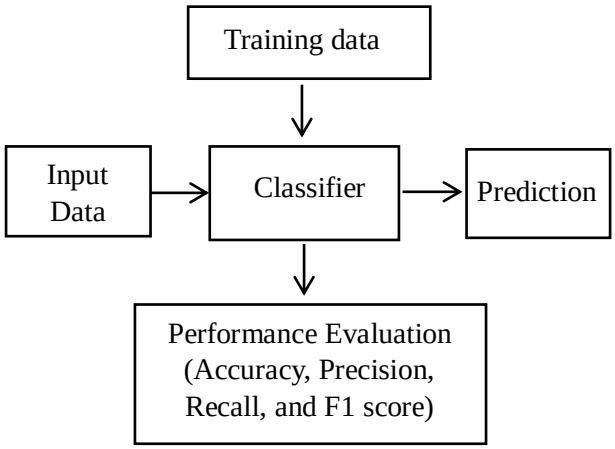

Fig. 1 Flowchart for the classification model.

II. LITERATURE REVIEW

In recent years, many scholars have done a great deal of research in this domain. They have implemented different classification models on the diabetes dataset used in this study to evaluate their performance. Therefore, this section aims to provide a brief analysis of the research works that have been proposed in this area.

Starting with Umair Munner Butt et al.'s [6] study, their team used three different classifiers, i.e., Random Forest (RF), Multilayer Perceptron (MLP), and Logistic Regression (LR). They also used long short-term memory, moving averages, and linear regression for predictive analysis. During the analysis, they found MLP outperforming other classifiers with an accuracy of 86.08% and long short-term memory improving the accuracy of significant prediction to 87.26%. Janhavi R. Raut et al. [7], however, used K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and RF for diabetes prediction. They evaluated the performance of the algorithm using correctly and incorrectly classified instances of the training dataset. Their team found the accuracy of KNN, SVM, and RF to be 71.35%, 73.43%, and 74.47%, respectively, concluding that RF is the best classification technique for diabetes prediction.

Apart from these studies, Farhana Bano et al. [8] used five different machine learning algorithms, namely SVM, Artificial Neural Network (ANN), Decision Tree (DT), LR, and Farthest First (FF). Their experimental results showed FF attaining superior correctness with an accuracy of 84.82%. However, in Olexandr Schamtko et al.'s study [9], they found LR to be the most effective classifier, with an accuracy of 78% among DT, LR, and KNN, while J.J. Khanam and Simoon Y. Foo's proposed work [10] implemented Neural Network (NN) alongside NB, SVM, LR, Adaboost, KNN, and DT. They used 1, 2, and 3 hidden layers in their neural network model, varying the epochs of 200, 400, and 800. The hidden layer 2 with 400 epochs provided 88.6% accuracy, which was the highest accuracy among other implemented models. Meanwhile, Deepti Sisodia and Dilip Singh Sisodia's experiment [11] evaluated the performances of DT, SVM, and NB using accuracy, precision, recall, and the f-measure score. Then, using those scores, they found NB to be most effective in predicting diabetes, with an accuracy of 76.30%.

All these past works clearly state that there is no particular classification model that can be totally considered efficient. In every group of algorithms used for comparison, there is a different algorithm that seems to stand out in correctly detecting the diabetes. Thus, there is a need to compare each classification algorithm on a step-by-step basis to identify new algorithms that can be suitable for predicting diabetes, which is what this paper is mainly about.

III. DATASET

The Pima Indian Diabetes Dataset (PIDD) used in this paper is taken from theUnited States National Institute of Diabetes and Digestive and Kidney Diseases. The dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases [12]. It has 768 records of both diabetic and non-diabetic individuals who are at least 21 years old. The individuals in this dataset are all females of Pima Indian heritage. The dataset has nine attributes: Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, Age, DiabetesPedigreeFunction, and Outcome. Out of these nine attributes, Outcome is the target variable, and the remaining eight are feature variables or medical predictors. The detailed

description of each attribute is provided in the Table I.

TABLE I

PIDD attributes and their descriptions and types.

Attributes	Description	Type
Pregnancies	Number of times a women is pregnant	Integer
Glucose	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	Integer
Blood Pressure	Diastolic blood pressure in mm Hg	Integer
Skin Thickness	Triceps skin fold thickness in mm	Integer
Insulin	2-Hour serum insulin in μIU/mL	Integer
BMI	Body mass index in kg/ m^2	Float
Diabetes Pedigree Function	Likelihood score of diabetes based on family history.	Float
Age	Age of Pima Indian females in years	Integer
Outcome	Class variable. '0' indicates non-diabetic, while '1' indicates diabetic.	Integer

In the dataset, there are some missing values for Glucose, Blood Pressure, Skin Thickness, Insulin, and BMI respectively. Therefore, those missing values are substituted with the average value of the respective attributes.

IV. CLASSIFICATION ALGORITHMS

This section provides a brief explanation of all four classification algorithms used in this study.

A. K-Nearest Neighbor (KNN)

KNN is a supervised machine learning algorithm that can be used for both regression and classification problems.

Since this paper uses KNN for classification purposes, first the algorithm finds the distance between the new data point and the existing training data points. The three most common methods KNN uses to calculate the distance are Euclidean, Manhattan, and Hamming[13]. After calculating distances, the algorithm chooses the K training data points that are closest to the given new data point. Out of those chosen data points, the algorithm finally picks the category or label that has the highest frequency in the group of K nearest neighbors.

In KNN, the value of *K* is always an integer and has a powerful effect on the performance of the algorithm. However, there is no magic way to find the optimum value of *K*. Therefore, for this study, a range of values for *K* is selected, and cross-validation is used to find the accuracy score for each value of *K*. After analyzing the accuracy score list, the optimum value of *K* was found to be 15.

B. Naïve Bayes (NB)

Unlike KNN, NB uses the following Bayes' theorem to find the label of given input data.

$$P(Y \mid X) = \frac{P(X \mid Y)\, P(Y)}{P(X)}$$

The theorem states that the probability of Xoccurring can be found given that Yhas occurred, where Y is a class or label variable and X is an n-dimensional feature vector that can be written as:

$$X = \{x_1, x_2, \ldots, x_n\}$$

x_1, x_2, ..., x_n represent the features of a training dataset that can be used for classification purposes, and when substituted for *X* and expanded using the chain rule, the probability equation above simplifies into:

$$P(Y \mid x_1, x_2, \ldots, x_n) = \frac{P(x_1 \mid Y) P(x_2 \mid Y) \ldots P(x_n \mid Y)\, P(Y)}{P(x_1)\, P(x_2) \ldots P(x_n)}$$

In NB, two assumptions are made regarding the features. One, they are assumed to be independent of each other, meaning the presence of one particular feature in the dataset does not affect the other. Two, each feature in the dataset has an equal effect on the outcome, meaning that one feature does not have more or less importance than another when making a prediction. For this study, there are only two class variables: diabetes positive or negative. However, in

multivariate classification, the class Y with the maximum probability is found.

Using the below function, we can find the class if the features are provided [14].

$$y = \text{argmax}_y \, P(Y) \prod_{i=1}^{n} P(x_i | \, Y)$$

The Naïve Bayes classifiers generally have higher accuracy and speed when implemented on larger datasets, but their performances are hindered when the features of the training data are dependent on each other.

C. *Support Vector Machine (SVM)*

SVMs are used for both classification and regression purposes, but they are widely used for dealing with classification problems. They create a best line or decision boundary, also called a hyperplane, to separate n-dimensional space into classes so that new data points can be conveniently classified into the correct categories. To create a hyperplane, SVM chooses extreme points known as support vectors. These vectors are closest to the hyperplane and affect the hyperplane's position.

There can be numerous hyperplanes or decision boundaries that can segregate the datapoint in SVM, so the hyperplane with the maximum margin, i.e., the maximum distance between the data point, is chosen. Such hyperplanes offer some reinforcement, which can categorize future data points more confidently.

Similarly, the hyperplanes are not always a straight line; they can also be a 2-D plane depending on the number of input features. In practice, SVM algorithms are executed using a kernel. The kernel takes a low-dimensional input space and converts it into a higher-dimensional space by adding more dimensions to the input space. This kernel trick can be used to implement more accurate SVM classifiers. In the context of this study, a linear kernel is used as it is found to be more effective in classifying data points compared to the other SVM kernels.

D. *XGBoost (XGB)*

XGBoost, or eXtreme Gradient Boosting, implements Gradient Boosted Decision Tree (GBDT) based on function approximation and uses several regularization techniques in addition to optimizing particular loss functions [15]. In XGB, the objective function is optimized using gradient descent, which finds the local minimum value of the function.

XGB is a separate library that needs to be installed on the system to utilize it, but in this study, instead of using the native XGBoost API, its Sklearn API is used to train the classifier first. Then the classifier is switched back to the native XGBoost classifier using the '***get_booster'*** method to access extra functionality. Similarly, while training the classifier, the objective is set to '***binary: logistic,'*** as there are only two classes in this study: diabetic or non-diabetic.

The main advantage of using XGB over other classification algorithms is that it supports parallel processing, which enables it to train models on a larger dataset in a reasonable amount of time and thus offer higher accuracy. Besides classification, XGB is also used for regression and ranking problems. However, XGB does not perform well on sparse and unstructured data. Since every classifier is required to correct the mistakes made by their predecessors learners, they are also susceptible to outliers.

V. CLASSIFIER ANALYSIS

In this study, four performance metrics, or measures, are implemented to evaluate the performance of each classifier: they are accuracy, precision, recall, and F1 score. The scores for each performance metric are calculated using the values of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) in the confusion matrix generated by the classification algorithms.

A brief explanation of each metric is provided below.

A. *Accuracy*

It is calculated by dividing the total number of correctly classified instances by the total number of classified instances. In terms of confusion matrix,

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

B. Precision

It is the model's actual correct prediction divided by the total prediction. In terms of confusion matrix,

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

C. Recall

It is the ratio between the numbers of positive instances correctly classified as positive to the total number of positive instances. In terms of confusion matrix,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

D. F1 Score

It is the harmonic mean of precision and recall. F1 Score is calculated using the following formula.

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

After calculating the scores for each performance metric using the above formulas, the results are represented in the bar diagram and table below.

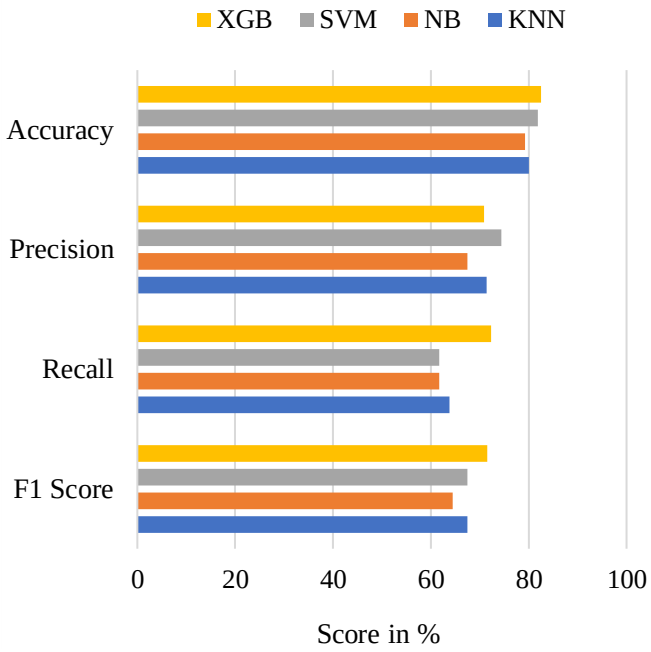

Fig. 2 A clustered bar diagram displaying the performancemeasures of each classifier.

TABLE II

The corresponding values (in percentage) of performance measures displayed in Fig. 2

	XGB	**SVM**	**NB**	**KNN**
Accuracy	82.47	81.81	79.22	80.10
Precision	70.83	74.36	67.44	71.43
Recall	72.34	61.70	61.70	63.83
F1 Score	71.58	67.44	64.45	67.42

The above results show that XGB performed best in terms of accuracy, recall, and F1 Score, recording the highest percentage values of 82.47%, 72.34%, and 71.58%, respectively. Similarly, in terms of precision, SVM performed well, with the highest score of 74.36%. It also recorded the second-highest scores for both accuracy and F1. NB, on the other hand, performed worse with the lowest score for each performance metric: 79.22% for accuracy, 67.44% for

precision, 61.70% for recall, and 64.45% for F1. Overall, the results show that XGB is the most effective classifier in predicting diabetes, followed by SVM in the group of KNN, NB, XGB, and SVM.

VI. CONCLUSION

Implementing classification algorithms to predict diabetes is one way to detect the disease correctly at the initial stage, but choosing the right classification algorithm is the major challenge. In the past, several studies on the same topic have been carried out using different classification algorithms. This study, however, compared four classification algorithms: KNN, SVM, NB, and XGB. Each algorithm performed differently on the dataset. XGB did best in predicting diabetes, with an accuracy of 82.47%, while NB performed worst. The second-best classifier was SVM. Still, there are many other classification algorithms that can be used for the prediction of diabetes in the future. Therefore, this study also invites future research on testing classifiers to predict diabetes by applying various dimensionality reduction strategies and comparing classifiers with different performance metrics than those used in this study.

REFERENCES

[1]*Diabetes*. Cleveland Clinic. Retrieved from https://my.clevelandclinic.org/health/diseases/7104-diabetes

[2] IDF Diabetes Atlas(10th ed.).(2021). International Diabetes Federation.

[3] *Diabetes*. (2023). WHO. Retrieved fromhttps://www.who.int/health-topics/diabetes

[4]J. Chaki, S.T. Ganesh, S.K. Cidham, S. Ananda Theertan. (2020). *Machine Learning and Artifical Intelligence-based Diabetes Mellitus Detection and Self-Management: A Systematic Review.* (pp. 3205-3206). Journal of King Saud University - Computer and Information Sciences (vol. 34).

[5] V. Sheth, U. Tripathi, A. Sharma. (2022). *A Comparative Analysis of Machine Learning Algorithms for Classification Purpose*. (p. 422). Procedia Computer Science (vol. 215).

[6]U.M. Butt, S. Letchmunan, M. Ali, F.H. Hassan, A. Baqir, and H.H. Raza Sherazi. (2021). *Machine Learning Based Diabetes Classification and Prediction for Healthcare Applications*. Journal of Healthcare Engineering (vol. 2021).

[7] J.R. Raut, Dr. Yogesh Sharma, Dr. Vinayak D. Shinde. (2020) *Performance Evaluationof Various Supervised Machine Learning Algorithmsfor Diabetes Prediction*. European Journal of Molecular & Clinical Medicine (vol. 7).

[8] F. Bano, Munidhanalakshmi K, Dr. R. Madana Mohana. (2021). *Predict Diabetes Mellitus Using Machine Learning Algorithms*. Journal of Physics: Conference Series.

[9] O. Shmatko, O. Korol, A. Tkachov, V. Otenko. (2021). *Comparison of Machine Learning Methods for a Diabetes Prediction Information System*. International Scientific and Practical Conference.

[10] J.J. Khanam, S.Y. Foo. (2021). *A comparison of machine learning algorithms for diabetes prediction*. ICT Express (vol. 7).

[11] D. Sisodiaa, D.S. Sisodia. (2018). *Prediction of Diabetes using Classification Algorithms*. Procedia Computer Science (vol. 132).

[12] *Pima Indians Diabetes Database*. Kaggle. Retrieved from https:///datasets/uciml/pima-indians-diabetes-database.

[13] A. Christopher. (2021). *K-Nearest Neighbor*. Medium. Retrieved from https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4.

[14] R. Gandhi. (2018). *Naive Bayes Classifier*.Medium. Retrieved from https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c.

[15] D. Levnits. (2018). *XGBoost Mathematics Explained*. Medium. Retrieved from https://dimleve.medium.com/xgboost-mathematics-explained-58262530904a.