Unraveling Diabetes Detection: A Comparative Study of Machine Learning Approaches

Jiten Agarwal
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: jitenagarwal05@gmail.com

Hardik Chauhan
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: hardik1057@gmail.com

Sneha Gupta
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: sneha9777gupta@gmail.com

Shreya Mondal
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: shreyamondal2362@gmail.com

Saumyadeep Mahanta
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: saumyadeepmahanta@gmail.com

Santos Kumar Baliarsingh
School of Computer Engineering
KIIT Deemed to be University,
Bhubaneswar, India 751024
Email: santos.baliarsinghfcs@kiit.ac.in

***Abstract*—The core focus of our work has been the utilization of Machine Learning (ML) to examine healthcare, with the main aim of recognizing early diabetes cases. Giving an early diagnosis of diabetes to prevent the growth of serious complications is the main task to achieve for a thorough treatment of the problem. Diabetes, being a chronic and enduring condition, presents a significant global challenge, being a primary contributor to premature mortality. We used ML and statistical methodologies to conduct a comparative evaluation of these modeling forecasts as an essential tool for diabetes diagnosis. While keeping the experimentation and evaluation rigorous, we study how successful the models are and their advantages and disadvantages within the healthcare field. We aim to disseminate findings that represent the practical applications within a real-world healthcare system. Moreover, it forms the foundation for individualized blood sugar diagnosis and therapy, ultimately enhancing mental well-being. By accomplishing these steps, personalized medicine could potentially offer a solution for diabetes treatment in the future.**

***keywords*—Diabetes, Machine learning, Comparative evaluation.**

I. Introduction

A. Background

Diabetes, a chronic metabolic disorder characterized by elevated blood sugar levels due to insufficient insulin production or utilization poses significant health risks worldwide. If left untreated, it can increase the likelihood of conditions such as cardiovascular disease, kidney damage, neuropathy, and vision impairment. While Type 1 diabetes typically develops in childhood or adolescence and necessitates lifelong insulin therapy, Type 2 diabetes, more prevalent among adults and often linked to obesity and lifestyle factors, can be managed through dietary modifications, exercise, medication, and insulin therapy as needed. Effective diabetes management entails vigilant monitoring of blood sugar levels, adherence to a strict diet, regular physical activity, and the timely administration of insulin or oral medications to prevent complications and maintain overall health.

In the contemporary era, Diabetes mellitus has emerged as a rapidly escalating health concern, particularly affecting various demographic groups. Early and timely detection of diabetes is essential for initiating effective interventions and preventing complications from arising. Conventional diagnostic methods often fall short in accurately identifying individuals at risk of developing the disease, highlighting the necessity to explore innovative techniques. ML has gained traction as an approach that leverages data-driven predictive patterns to enhance diabetes diagnostics.

According to the latest data from the World Health Organization (WHO)[**?**], over 422 million individuals worldwide are living with diabetes, with projections indicating a further increase in these numbers. Early diagnosis is crucial for diabetes prevention and control, yet some traditional diagnostic methods lack the precision needed to identify individuals at risk. Advanced data mining processes are used by ML to quicken diagnosis and enhance healthcare quality. ML algorithms consider various factors like medical history, genetic predisposition, lifestyle, and laboratory results to identify those at risk of diabetes. These algorithms can detect patterns and relationships by analyzing vast amounts of data, enabling proactive screening and intervention. Moreover, ML enhances personalized medicine by providing doctors with patient-specific prognoses and tailored treatment plans. However, ML-based diabetes diagnosis has challenges, such as incomplete datasets, feature linkage issues, and result interpretability. Researchers are actively exploring ML approaches, including deep learning, to address these challenges and enhance accuracy and efficiency in diabetes detection. A comparative study of ML models for diabetes detection is warranted to assess their accuracy and performance. Evaluation criteria include sensitivity, specificity, and predictive accuracy. Such a study aims to provide insights into the strengths and limitations of existing ML methods for diabetes detection, guiding the development of new techniques or improvements to existing approaches for better patient outcomes.

B. Motivation

The motivation driving the research is the pressing need to overcome obstacles that obstruct the diagnosis of diabetes. Despite advancements in health technology, the persistent issue of missed diagnoses and delayed detection continues to incur significant health and economic costs. Against this backdrop, our research embarked on a mission to analyze and compare various ML algorithms to identify the most effective approach for diabetes detection. We aimed to develop methods that not only provided higher accuracy but also ensured interventions were delivered without delay.

Our study sought to bridge the gap between technology and healthcare. By leveraging ML, our goal is to equip physicians with advanced tools that facilitate early disease identification. This entails harnessing data and uncovering correlations that might otherwise go unnoticed, leveraging the strengths of algorithms. Ultimately, our initiative aimed to provide impactful insights that empower healthcare providers to make informed decisions.

In short, our journey was one of empowerment−empowering clinicians to take proactive measures, empowering patients to pursue wellness with determination, and empowering humanity as a whole in the battle against diabetes.

C. Contributions

This report contributed to the healthcare analytics field with a comprehensive assessment of the performance of different ML models in the diagnosis of diabetes. From a theoretical standpoint, the study not only explored the basics of the predictive algorithms that have been used but also validated their implementation using empirical data, thus capturing the performance of diverse models such as Naive Bayes (NB) [**?**], Support Vector Machines (SVM) [**?**], Decision Trees (DT) [**?**], k-Nearest Neighbours (kNN) [**?**], and Random Forest Classifier (RFC) [**?**]. We brought together findings obtained by analyzing real-world data and provided practice-centered advice, which may additionally be used to create evidence-informed guidelines. Moreover, in addition to the current study, our research is a basement for yet-to-be-investigated areas in predictive analytics, as this will lead to the establishment of innovations and advancements in diabetes management strategies. Along with the concreteness of certain types of ML algorithms, it is important to highlight that they are both fully applicable and effective in the medical domain. This approach, therefore, ensures that the findings we get are both credible and sensible to make regarding the current developments in healthcare analytics.

II. Literature review

Sisodia *et al.* [**?**] deployed three ML algorithms involving SVM, NB and DTs to forecast the outbreak of diabetes. The authors achieved 76.30% accuracy by using the NB classifier with the Pima Indians Diabetes Dataset [**?**].

Rastogi *et al.* [**?**] utilized different ML algorithms like RFC, K- means clustering, Linear regression and Logistic Regression, DTs, SVM and NB for the forecasting of diabetes. The scholars found that logistic regression was a more reliable tool than the other ML techniques, with an accuracy of 82.46% with the diabetes dataset from Kaggle.

Daanouni *et al.* [**?**] implemented an ML algorithm on the dataset of the Pima Indians Diabetes Dataset for the prediction of diabetes. This was achieved by experimenting with artificial neural networks, kNN, deep neural networks, and DTs on the data set. They concluded that in the Deep Neural Network, feed-forward network was the best, which was associated with the highest accuracy rate of 90%.

Xue *et al.* [**?**] selected Support Vector Machine (SVM), NB, and Light Gradient-Boosting Machine (LightGBM) ML algorithms. The authors highlighted the importance of considering various approaches when distinguishing successful prediction strategies. The Sylhet Diabetes Hospital in Bangladesh gave this dataset.

Dutta *et al.* [**?**] pointed out that the classifiers used include NB, Random Forest Classifier (RFC), Decision Tree (DT), Extreme Gradient Boosting (XGBoost), and LightGBM.The most definitive finding was achieved by employing logistic regression, which has a 96% record of accuracy acquired from a just-released DDC dataset in Bangladesh.

Zou *et al.* [**?**] believed that early detection of the disease could assist physicians in managing patient conditions and improving clinical decision-making. In this case, he provided diabetes prediction in the model through RF, DTs, and deep learning networks. The information furnished was gathered from the physical examination records of patients in a Chinese hospital called Luzhou. The authors studies in combination pointed to an accuracy rate of RFC standing at 80.84%.

Soni *et al.* [**?**] involved six ML models: kNN, SVM, DT, Logistic Regression, RFC, and Gradient Boosting, which were used to predict diabetes in the case of the Pima Indian Diabetes Dataset. The best-performing method was RFC, whose accuracy function was 77%.

Muhammad *et al.* [**?**] presented in the year 2020 explored classifiers such as RFC, gradient boosting, kNN, SVM, logistic regression, and NB. Based on the Mohamed Specialist Hospitals data from Nigeria, the writer coined this name. Unlike algorithms such as Gradient Boosting (82%), kNN(75%), NB (86%), Logistic Regression (90%), and RFC (90%), we concluded that RFC was still the one with the highest percentage of accuracy (87%).

K. VijiyaKumar *et al.* [**?**] utilized ML models to predict diabetes, with a focus on RFC. They tested the accuracy of their model using type 2 diabetes data, achieving an accuracy of approximately 90%. Their approach demonstrated superior performance compared to other machine learning algorithms for diabetes prediction.

L. V. R. Kumari *et al.* [**?**] used ML methods to enhance the accuracy of early diabetes prediction. They incorporated algorithms like NB, kNN, Logistic Regression and RFC. Among these algorithms, kNN algorithm gave a high accuracy of 78.57%.

N. Fazakis *et al.* [**?**] developed a model utilizing ML technologies. Initially, individual classifiers such as Naive

Bayes (NB), Decision Trees (DTs), Random Forests (RFs), and Logistic Regression were employed independently. Subsequently, ensemble learning techniques including majority voting (weighted or unweighted) and stacking were utilized for type 2 diabetes prediction. The ensemble WeightedVotingLRRFs ML model achieved an Area Under the ROC Curve (AUC) of 0.884, suggesting its effectiveness in enhancing diabetes prediction, particularly with weighted voting. Additionally, the study compared various ML models using both inductive and transductive learning approaches, along with the Leicester and Finnish Diabetes Risk Score systems. Data analysis was conducted using the English Longitudinal Study of Ageing (ELSA) database.

M. A. R. Refat *et al.* [**?**], explored several machine learning including XGBoost, RF, DTs, SVM, kNN and Logistic Regression and deep learning classifiers such as Multilayer Perceptron, Artificial Neural Network, Long Short-Term Memory(LSTM) for early prediction of diabetes. They utilized diabetic dataset with 17 features, including the class label, sourced from the UCI repository. The results indicated that while most algorithms achieved an accuracy of 90%, the XGBoost algorithm demonstrated significantly better performance, achieving 100% accuracy.

III. Proposed Work

The prevalence of diabetes was the issue at hand, as evidenced by the loss of numerous lives, yet it remained a condition for which a complete cure had yet to be found. Sometimes diabetes is discovered too late after many years, and the patient may live with its symptoms without even knowing about it. A lot of progress has been made in building the system to diagnose diabetes, and indeed, more efforts have become relevant with the ever-evolving Artificial Intelligence (AI). Specifically, our study focused on the implementation of five mainstream Machine-learning algorithms on three repositories of datasets and the inspection of accuracy performance indicators for all the ML algorithms on all three datasets.

A. Preprocessing

1) Data Collection: The primary and most crucial step in constructing any ML model is data collection. The accuracy score of the algorithms used are derived from the sample used. The data provided by the three datasets [**?**], [**?**], [**?**] on diabetes at Kaggle has been collected, and we trained our model on this data to predict whether someone has diabetes or not.

2) Data Standardization: From sklearn, we employed the StandardScaler object from the preprocessing module. For this, it cuts down on standardization by taking the mean out and dividing it through to get a unit variance as shown in Equation 1.

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

If x denotes the data point, μ represents the mean of the dataset, and σ signifies the Standard deviation of the dataset.

A fit method of the sklearn library is used to obtain the mean and standard deviation of the data. The process then involves the use of the transformation method, with the mean and standard deviation from previously calculated values deducted from it.

3) Data Splitting: The generic data has been split into train and test data where the split uses the 80-20 rule, therefore, 80% of the data is used as training data, and the remaining 20% of the data is used as testing data.

B. Models Used

1) Naive Bayes: In the present study, the Gaussian-Naive Bayes (GNB) variation of the NB classifier is utilized, as it fits the continuous data well and uses the idea that all features have a normal distribution. In light of the strict conditionality of independent features, the probabilistic classification classifier NB applies the Bayes theorem. The algorithm hypothesizes the presence of one feature of a certain group as unbiased and independent of the other features; that is why it is named NB. When we want an immediate, responsive action, this is when the algorithm tends to be used more often. The model's performance is considered by determining its accuracy and drawing a confusion matrix with the help of Table II. The mathematical formula for Bayes Theorem is shown in Equation (2)

$$P(A \mid B) \ = \frac{P(B \mid A) \cdot P(B)}{P(A)} \tag{2}$$

Where:
$P(A \mid B)$ is the posterior probability of class (A) given predictor (B).
$P(A)$ and $P(B)$ are the probabilities of class and predictor respectively.
$P(B \mid A)$ is the likelihood which is the probability of the predictor given class.

2) Support Vector Machine: It can be used for linear and non-linear classification, regression, and also detecting outliers. The primary goal of this strategy is to define a hyperplane that separates N-dimensional space into distinct classes. The algorithm tries to find such a hyperplane that shows the most succeeding separation margin between two classes. It decides on a hyperplane whose distance from it to the nearest data point on both sides is maximum. And if such a hyperplane is present it can be called the maximum margin hyperplane. SVMs are, in general, designed to perform a lot of different tasks such as image detection, spam detection, face detection, etc.

3) Decision Trees (DT): This algorithm decides choices based on questions and a criteria framework. The approach is just like a flowchart, where a node means a feature, a branch means a decision, and a leaf points to the result.
Components of a DT:-

a) Root Node: This node extends from the top or starting point of the decision tree, where all the features are divided into different branches based on various conditions.

b) Decision Nodes: These nodes, which are also called decision nodes, are constructed by splitting the root node. They represent intermediate decisions.

c) Leaf Nodes: The last nodes of the tree from where even further splitting does not happen. Furthermore, they give concrete results.

The algorithm is composed of recursion in such a way that it rives the data set into minute subsets. Each node is evaluated, and the algorithm determines which feature to split the data. In our algorithm, the maximum depth is 2 means we will use a maximum of 2 levels in our tree. Moreover, this reduces the model's tendency to pick up only specific data patterns, known as overfitting, ensuring that the resultant tree is not overly complex. The criterion used is 'entropy' and it is calculated as shown in Equation (3)

$$E(S) = -\sum_{i=1}^{n} p_{C_i} \log_2(p_{C_i}) \tag{3}$$

Where $E(S)$ is the entropy of dataset S, p_{C_i} is the proportion of instances in S that belong to class C_i and n is the number of classes. It is used for the information gain. Information gain is the reduction in entropy or surprise by splitting a feature and it is calculated as shown in Equation (4)

$$Gain(S, A) = E(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} E(S_v) \tag{4}$$

where S is the total sample space, A is the attribute, $Values(A)$ represent the set of all possible values of an attribute A and S_v represents the subset of S for which attribute A has value v.

4) k-Nearest Neighbours (kNN): The approach is fairly easy, straightforward, and accurate based on the samples that are as close or the closest one to the reference as possible, known as the neighbors, where the closeness is calculated using Euclidean Distance, Manhattan Distance, etc. This specific model will assess the five nearest neighbors for its predictions (k = 5) and will use Euclidean distance as shown in Equation (5).

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} \tag{5}$$

Where p and q are the two points in the data collection. p_i and q_i are the i^{th} coordinates respectively, and n is the number of dimensions or features in the dataset.
It does not require any assumptions about the data, this helps the algorithms to adapt to different patterns and make predictions based on the local structure of data. The choice of 'k' significantly influences the performance of the algorithm. A low value of k can lead to overfitting, while a high value of k can increase bias and result in high test error. Additionally, k should be an odd number to prevent ties and ensure there is always a majority class.

5) Random Forest Classifier (RFC): It is a powerful machine-learning algorithm that works by creating several DTs during the training phase. While predicting, the algorithm totals the results of all trees; for classification tasks, the preferred method of decision-making is through voting, while for regression tasks, the preferred method is through averaging. It predicts the output with high accuracy for large datasets. RFCs are widely used for classification, regression, and reduced overfitting. In our model, we designed an RFC with estimators=100 to create a maximum of 100 trees, max_depth=100 which makes sure that the maximum depth of any tree generated is not more than 100, min_samples_split=20 which specifies the minimum number of samples required to split an internal node which here is set to 20, the min_samples_leaf which defines the minimum number of samples required for a node to be considered a leaf is one, max_features is set to the square root of the number of features required to consider when looking for the best split. The random state is set to 5, which makes the output deterministic.

IV. Experiment

A. Dataset

A large dataset is needed to undertake a comparative study of various features and models. Here, we have used three datasets from different sources and with different numbers of features in the data set. We sourced the data sets from Kaggle, making them more resilient and versatile.

1) Diabetes Dataset: [**?**] This Dataset was sourced from Kaggle and has 769 entries and 9 features namely Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, and Outcome.

2) Diabetes prediction dataset: [**?**] This dataset was sourced from Kaggle and has 100001 entries and 9 features namely Gender, Age, hypertension, heart_disease, smoking_history, BMI, HbA1c_level, blood_glucose_level, and diabetes.

3) Diabetes Health Indicators Dataset: [**?**] This Dataset was sourced from Kaggle and has 253681 entries and 22 features namely Diabetes_012, HighBP, HighChol, CholCheck, BMI, Smoker, Stroke, HeartDiseaseorAttack, PhysActivity, Fruits, Veggies, HvyAlcoholConsump, AnyHealthcare, NoDocbcCost, GenHlth, MentHlth, PhysHlth, DiffWalk, Sex, Age, Education, and Income.
This has been summarized in the table I.

Table I
Dataset

Name of dataset	Features	Target Class
Diabetes Dataset	9	2
Diabetes prediction dataset	9	2
Diabetes Health Indicators Dataset	22	2

B. Experiment setting

In this study, we have compared machine-learning algorithms on various datasets to predict diabetes.

The initial step involved the Standardisation and Label encoding of the datasets, this is achieved by using StandardScaler() and LabelEncoder() function. Following this, the datasets were split using an 80-20 split i.e. 80% of the data was used to train the model, and the rest was used to test the model to calculate the accuracy.

Subsequently, kNN, RFC, DT, NB and SVM were applied to the datasets, and their performance was evaluated using the accuracy score and confusion matrix that are formulated using equation (6) and Table II

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{6}$$

Table II
CONFUSION MATRIX

	Predicted	**Predicted**
Actual	True Positive (TP)	False Negative (FN)
Actual	False Positive (FP)	True Negative (TN)

Where

True Positives (TP): When we predicted that an event happened and it did occur, those are considered True Positives (TP).

True Negatives (TN): We predicted that there would be no event, and indeed, no event occurred.

False Positives (FP): We predicted yes, but the event didnt happen. Also known as Type I error.

False Negatives (FN): We predicted no, but the event did happen. Also known as Type II error.

Subsequently, the accuracy scores derived from these models are represented in a barplot, providing a visual interpretation of the model's performance. This graphical representation facilitates a more intuitive understanding of the distribution and variance of accuracy scores across different models.

V. RESULTS AND DISCUSSION

A. Analysis of Algorithms on the Datasets

In this section, we compared the accuracy of each algorithm using five models: NB, SVM, DT, kNN, and RFC on three different datasets.

1) Diabetes Dataset: For this dataset, RFC was the best fit for predicting Diabetes with its accuracy score of 81.2% and confusion matrix is shown in Table III.

Table III
CONFUSION MATRIX FOR DIABETES DATASET

	Predicted Diabetes	Predicted Not Diabetes
Actually Diabetes	86	13
Actually Not Diabetes	16	39

2) Diabetes prediction dataset: For this dataset, DT was the best fit for predicting Diabetes with its accuracy score of 97.2%, and the confusion matrix is shown in Table IV.

Table IV
CONFUSION MATRIX FOR DIABETES PREDICTION DATASET

	Predicted Diabetes	Predicted Not Diabetes
Actually Diabetes	18299	0
Actually Not Diabetes	551	1150

3) Diabetes Health Indicators Dataset: For this dataset, DT and SVM were the best fits for predicting Diabetes with an accuracy score of 86.1%, and their confusion matrix is shown in Table V.

Table V
CONFUSION MATRIX FOR DIABETES HEALTH INDICATORS DATASET

	Predicted Diabetes	Predicted Not Diabetes
Actually Diabetes	43671	0
Actually Not Diabetes	7065	0

4) Analysis of implemented algorithms on the datasets: As observed the performance of five different ML models: kNN, SVM, NB, RFC, and DT on three different datasets (Diabetes Dataset, Diabetes prediction dataset, Diabetes Health Indicators Dataset.

kNN performed consistently across all datasets with the highest performance on Diabetes prediction dataset.

SVMs also showed consistent performance across all datasets, slightly outperforming kNN on Diabetes prediction dataset.

NB had the lowest performance on Diabetes prediction dataset and Diabetes Health Indicators Dataset compared to the other models. However, its performance on Diabetes Dataset was comparable to that of SVM.

RFC performed well on all datasets, with its best performance on Diabetes prediction dataset.

DT had the highest performance on Diabetes prediction dataset and Diabetes Health Indicators Dataset, but its accuracy dropped on Diabetes Dataset.

These observations could help in choosing the right model based on the dataset at hand. However, it is important to consider other factors such as the nature of the data, the interpretability of the model, and the computational resources available. These results are summarised in Table VI

Table VI
ACCURACY SCORES ON ALL THE DATASETS

Name of Dataset	kNN	SVM	NB	RFC	DT
Diabetes Dataset	0.805195	0.779221	0.772727	0.811688	0.733766
Diabetes prediction dataset	0.96110	0.96130	0.90380	0.97055	0.97245
Diabetes Health Indicators Dataset	0.847406	0.860750	0.774539	0.859094	0.860750

VI. GRAPHICAL COMPARISON OF CLASSIFICATION MODELS ON THE DATASETS

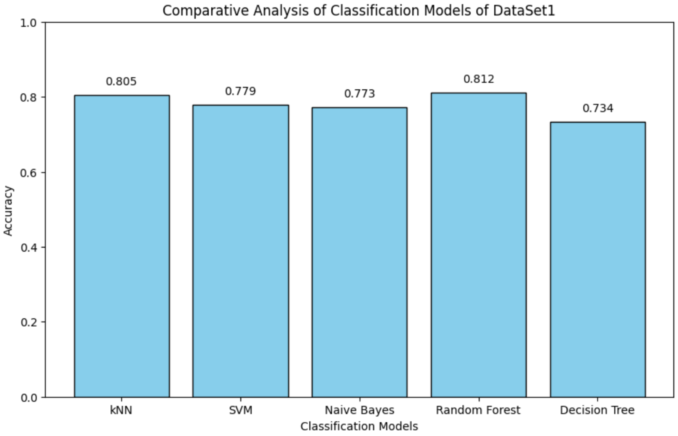

Figure 1. Analysis of all algorithms on Diabetes Dataset

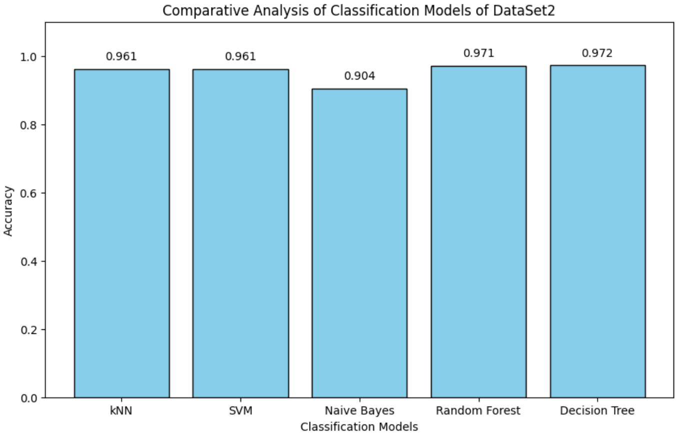

Figure 2. Analysis of all algorithms on Diabetes prediction dataset

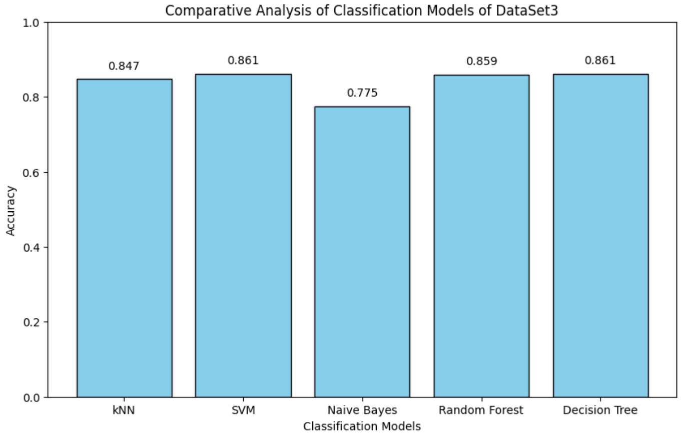

Figure 3. Analysis of all algorithms on Diabetes Health Indicators Dataset

VII. CONCLUSION AND FUTURE WORK

It has been identified that this study critically analyzed five ML techniques used in predicting diabetes: kNN, DT, RFC, SVM, and NB. Hence, we deduced the effectiveness as well as the performance of these techniques through the application of them to the diagnosis of diabetes. Lastly, our analysis concluded that from Table VI the RFC was the most probable model for diabetes prediction since it showed better predictive accuracy and robustness compared to the other models. However, the outcomes of each model was different when the overall scores were measured which is shown in Figure 1, Figure 2, and Figure 3, which means that diabetic predictors should be chosen based on various parameters.

In addition to this, this research also contributed to the whole knowledge base regarding diabetes prediction by showing how the ML approaches could be applied to improve early detection and management of this disease. Similarly, these models could assist healthcare workers with better choices and interventions to reduce the likelihood of diabetes complications.

Soon, the use of ML algorithms will enable us to envision an era of diabetes prediction. To boost forecast capability as well as resilience, one can start with the ensemble learning approach, which involves a manifold of models with the best features. Boosting model performances and generalizations through ensemble methods such as stacking, boosting, and bagging is very successful in several instances. By including and combining development-specific attributes and data sources like environmental factors, lifestyle habits, or genetic biomarkers, the pathophysiology of diabetes can be elucidated. ML algorithms can process complex relationships and patterns thanks to finding meaning and interconnecting different data sets. It enables advanced forecasting and tailored measures to be used.

References