Predicting Diabetes Mellitus with Machine Learning Techniques

Heba Ahmed Jassim[1], **Omar R. Kadhim**[2], **Zahraa Khduair Taha**[3], **Johnny Koh Siaw Paw**[4], **Yaw Chong Tak**[5], **Tiong Sieh Kiong**[6]

[1]Computer Engineering Department, Al-Iraqia University, Baghdad, Iraq
hiba.a.jasim@aliraqia.edu.iq

[2]Network and Cybersecurity Department, Al-Iraqia University, Baghdad, Iraq
omarraad@aliraqia.edu.iq

[3]College of Graduate Studies, Universiti Tenaga Nasional, Kajang, Malaysia
PE21268@student.uniten.edu.my

[4]Institute of Sustainable Energy, Universiti Tenaga Nasional, Kajang, Malaysia
johnnykoh@uniten.edu.my

[5]Institute of Sustainable Energy, Universiti Tenaga Nasional, Kajang, Malaysia
chongty@uniten.edu.my

[6]Institute of Sustainable Energy, Universiti Tenaga Nasional, Kajang, Malaysia
siehkiong@uniten.edu.my

Article History

Received: Mar. 10, 2025

Revised: May 26, 2025

Accepted: Jun. 13, 2025

Abstract

Blood sugar issues are a major health issue worldwide, with their incidence growing rapidly and affecting human health, economic systems, and societal structures. If diabetes remains untreated and undiagnosed, it can cause blood sugar levels to vary significantly, potentially damaging essential organs like the kidneys, eyes, and arteries of the heart in critical cases. As a result, there is an increasing focus on the prevention and early detection of diabetes mellitus within the medical community. Utilizing machine learning algorithms to analyze appropriate datasets for early disease prediction could prove life-saving. The objective of this paper is to examine four algorithms that are proposed to enhance the diagnosis of diabetes. This research analyzes the effectiveness of various machine learning algorithms in processing datasets with minority classes. The evaluation was based on the classification report (including accuracy, precision, recall, and F1-score), the confusion matrix, and the ROC AUC. The Diabetes Prediction Dataset is used to evaluate four machine learning algorithms. The classifier that deserves a singular mention is the Artificial Neural Network (ANN), which achieves a 97% accuracy rate. This demonstrates its capability of classifying instances that are common and less common types. The Random Forest and Decision Tree models also perform well in terms of their ability to deliver strong performance, and the outcome shows some incremental differences, suggesting their ability to manage the dataset is quite high. However, the Support Vector Machine (SVM) model performs worse than all the above models at 96.36% and seems to struggle with the correct classification of less frequent instances. Therefore, it would be problematic to distinguish between classes that are prominent and those that are not. Notably, the ANN, Random Forest, and Decision Tree models effectively identify cases that are more likely to capture rare cases, an important aspect when dealing with datasets that have class imbalance.

Keywords- Diabetes prediction, Neural network, Random Forest and Support Vector Machine.

I. INTRODUCTION

Diabetes mellitus, or simply diabetes, describes a group of metabolic diseases that result in high blood sugar levels, also known as hyperglycemia. As time goes by, it can lead to severe and intricate health and systemic complications, including renal failure, heart attack, apoplexy, peripheral arterial disease, cardiovascular disease, and damage to the micro and macro blood vessels and nerves. Research by the Canadian Diabetes Association indicates that in Canada, the number of diabetes patients will grow from 2.5 million in

http://doi.org/10.58564/IJSER.4.2.2025.315

the year 2010 to 3.7 million patients by 2020. The growing number of diabetes cases is a global problem, not limited to Canada alone [2]. Currently, in Thailand, a third of people with diabetes are undiagnosed, bringing to light serious worries about the diabetes epidemic. Observations also show a trend towards younger individuals being diagnosed with the condition.

Furthermore, women are found to be more affected by diabetes than men, and the disease is more frequent among obese individuals than those of normal weight. These insights are driving interest in disease classification research, which holds considerable promise for improving healthcare and tailoring treatment to individual needs [3]. However, the existing models face several issues, such as limited classification performance, poor generalization [3], and difficulties in capturing imbalanced data [4].

This study aims to develop a technique for predicting diabetes risk in an individual without blood samples and hospital visits, which enhances health awareness and promotion. Furthermore, this research aims to develop an online diagnostic application, which is expected to be developed to be easily usable by the general population. It is important to note that this application is intended primarily for the purpose of preliminary identification of possible patients only. Individuals who, after screening, are considered to be at risk of diabetes are advised to get a medical evaluation from a qualified doctor for timely and appropriate management of diabetes-related complications.

The arrangement of this paper is as follows: It opens with the section: "Literature Review", which gives details about the literature central to the methods of developing the prediction models and background information. Next, the "Methodology" section outlines the strategies implemented to plan and carry out the study. The "Experimental Results" section then presents a concise overview of the outcomes obtained. These findings are further analyzed in the "Discussion" section, highlighting major insights, areas for potential improvement, and any constraints identified during the study. The paper wraps up with the "Conclusions" section, summarizing the endpoints of the research.

II. LITERATURE REVIEW

Kandhasamy *et al.* [4] utilized different machine learning classifiers, including J48 Decision Tree, K-Nearest Neighbors, Random Forest and Support Vector Machines were compared in the study of diabetes mellitus patients identification. To evaluate the performance of these classifiers, data samples coming from the UCI machine learning repository were used for assessing the diagnostics accuracy/accuracy of detection/classification for reviewing the condition.

A study by Nai-Arun *et al.* [3] specifically investigated four popular classification models: Artificial Neural Networks, Decision Tree, Logistic Regression and Naive Bayes. After this initial assessment, the work investigated Bagging and Boosting methods to enhance these classification models' performance and reliability. In this research, Ribeiro et al. [5] proposed a data classification methodology that improves the efficiency of input preprocessing by reducing data redundancy using well-known ICA methods like JADE, Fast ICA, and INFOMAX. A one-class SVM classifier was used for diabetic/non-diabetic classification. The efficiency of this classification was tested using a combination of non-invasive and invasive measures.

In this regard, Hayashi and Yukita, in their [6] study, proposed an example of clear, understandable, and precise classification rules from the Pima Indian Diabetes (PID) dataset through a new rule extraction algorithm, Re-RX, coupled with J48graft, combined with the selection techniques of sampling (sampling Re-RX with J48graft. The research in Mercaldo *et al.* [7] proposed a method using a machine learning algorithm able to distinguish between diabetic and non-diabetic patients. They evaluated their method on real-world data extracted by the Pima Indian population near Phoenix in Arizona. Indoria *et al.* [8] focused on recent machine learning developments, which have made major impacts on diabetes detection and diagnosis. Nilashi *et al.* [9] used machine learning techniques, a new hybrid intelligent system for diabetes disease classification. To cluster the experimental diabetes disease dataset, they applied EM clustering algorithm. and for the classification of disease types, they use the SVM algorithm. In addition, PCA was used to reduce dimensionality and address multicollinearity in the dataset. Lukmanto *et al.* [10] used FS to find the useful features in the dataset and to train the dataset with the aim of generating the fuzzy rules, using SVM at last, and classifying the output through a Fuzzy inference process. Rawat *et al.* [11] introduced five machine learning techniques to analyse and predict Diabetes Mellitus: Logic Boost, AdaBoost, Naive Bayes, Robust Boost, and Bagging. Deo *et al.* [12] aimed to evaluate the effectiveness of various classifiers in predicting patient disease probabilities with high precision and accuracy. They conducted experiments using nine attributes sourced from the UCI Repository, applying several classification algorithms: Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), K Nearest Neighbor (KNN), Naive Bayes (NB), and Logistic Regression (LR)—on the Pima Indians Diabetes dataset.

Divakar *et al.* [13] developed a model that can effectively predict diabetes to enable early detection of the disease. Two classification algorithms were used in their work: the Support Vector Machine and the Fine Decision Tree. Maniruzzaman *et al.* [1] applied a feature selection mechanism to give the highest classification accuracy. In order to verify their hypothesis, they developed a machine learning system that combines the Logistic Regression and Random Forest. YE *et al.* [14] employed a variety of techniques in NLP and ML, including the use of a CNN with both word embeddings and UMLS entity embeddings. Kumari *et al.* [15] employed a feature selection with the help of a neighborhood search method in order to get an effective subset of features that enhances classification performance. Then, a feature ranking model refines the selected features. Finally, a neural network classifier was trained to classify these features. Mainenti *et al.* [16] attempted to apply machine learning algorithms to classify various types of diabetes mellitus based on clinical data collected from patients with diabetes through standard hospital routines. Bashir *et al.* [17] considered two Pima Indian diabetes datasets to analyze different cases of diabetes mellitus. They indicated that several machine and deep learning approaches had already been applied in the past to these types of Pima Indian Diabetes datasets, and hence a number of highly successful diagnostic tools for diabetes had come into existence. They have also identified the research gap in the application of varied techniques within the biomedical field. Zhou *et al.* [18] proposed to diagnose current health conditions and estimated the risk of diabetes, treating it as a classification problem. The approach in that work was inspired by the model which is reliant on the hidden layers of a deep neural network along with dropout regularization to avoid overfitting. Abiyev *et al.* [19] suggested a diagnosis of diabetes using a neural network together with a Type-2 fuzzy system. Using statistical data, they used the framework of a Type-2 fuzzy neural network (T2FNN) to evaluate T2FNN for the diagnosis of diabetes.

Alyoubi *et* al. [20] designed fully automated diagnostic systems that outperform conventional manual techniques in terms of reducing errors and saving time and costs, thus reducing manpower. These systems classify images of Diabetic Retinopathy into five stages-no-DR, mild, moderate, severe, and proliferative DR with high efficiency, and help identify the exact site of lesions on the retina. NAHZAT *et al.* [21] used the Pima Indian Diabetes Dataset to implement several machine learning classifiers such as Random Forest (RF), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), Decision Tree (DT), and Support Vector Machine (SVM) for forecasting diabetes. The accuracy of the performance varied among the different models used. Dudkina *et al.* [22] developed a machine learning model based on decision tree techniques. Aparicio *et al.* [23] carried out a systematic review across 90 studies to explore major opportunities for improving diabetes prediction through machine learning. They evaluated eighteen different models, finding tree-based algorithms to be the most effective. Despite the ability of Deep Neural Networks to process large and noisy datasets, their performance was not as strong in this specific application. Table I summarizes different studies associated with diabetes disease.

TABLE I. LISTS DIFFERENT PUBLICATIONS THAT ADDRESS DIABETES MELLITUS WITH DIFFERENT ALGORITHMS

Reference	Dataset	Algorithms	Feature Processing	Highlights	Research Gap
[4]	UCI Repository	J48, KNN, Random Forest, SVM	Non	Basic ML classifiers compared	Lack of feature engineering
[3]	Not explicitly stated	ANN, Decision Tree, Logistic Regression, Naive Bayes	Non	Improvement with ensemble methods	No mention of feature selection and only
[5]	Non	One-class SVM	ICA (JADE, FastICA, INFOMAX)	Efficient preprocessing for classification	Focus on binary SVM
[6]	Pima Indian Diabetes dataset	Re-RX with J48graft	Rule extraction	Explainable AI with rule generation	lacks an evaluation hybrid model
[7]	Real-world PID data (Phoenix, Arizona)	Not explicitly named	Non	Real-world dataset	No performance metrics to evaluate
[8]	General overview	Recent ML algorithim	Non	Review of ML developments in diabetes	no empirical model evaluation
[11]	Non	Logic Boost, AdaBoost, Naive Bayes, Robust Boost, Bagging	Non	investigate multiple ensemble models	non
[12]	UCI dataset	SVM, DT, RF, KNN, NB, LR	Nine featuers	Comparison of multiple ML models	No advanced preprocessing or ensemble integration
[13]	Non	SVM, Fine Decision Tree	Non	Early detection risk	restricted algorithmic field
[1]	Non	Logistic Regression, Random Forest	Feature Selection	Highest accuracy classification	Lacks validation hybrid model
[14]	Non	CNN with NLP techniques	Word + UMLS embeddings	Use of embeddings in classification	Misses structured features.
[15]	Non	Neural Network	The relevant feature is selected and ranked.	Refined feature subset via neighborhood search	Lacks benchmark
. [16]	clinical dataset	Non	Non	Applied ML to hospital routine data	Models not specified
[17]	Two Pima Indian datasets	ML and DL algorithms	Non	Historical review of successful models	No implementing new models.

[18]	Non	DNN / dropout	Integration Deep learning with regularization	Risk levels classification	No update models
[19]	Statistical data	Type-2 Fuzzy Neural Network	Fuzzy logic integration	Integration of NN and fuzzy logic	High computation load
[20]	Retinal images	CNN	classification	Classifies DR stages with high efficiency	not specified for predicting diabetes
[21]	Pima Indian Diabetes Dataset	RF, KNN, ANN, DT, SVM	Non	accuracy evaluation for comparison	No advanced preprocessing applied to features.
[22]	Non	Decision Tree	Non	ML model based on DT	No benchmark for evaluation
[23]	90 datasets	18 ML algorithms	Non	Tree-based algorithms are used	misses structured features

III. METHODOLOGY

Figure 1 below displays the diagram of the proposed model, outlining the step-by-step flow and sequence of the experimental procedure conducted in this study. The first step of the methodology is preprocessing, which includes cleaning the dibets dataset by identify dublicate and missing values. Duplicate records are selected and removed to prevent bias toward redundant data. One-hot encoding is used to transform data into numerical format, then standardize the data to normalize the feature scales and enhance the convergence of models. After preprocessing, four machine learning algorithms, Artificial Neural Network (ANN), Random Forest (RF), Decision Tree (DT), and support vector machine (SVM), are applied to classify patients into diabetics and non-diabetics.

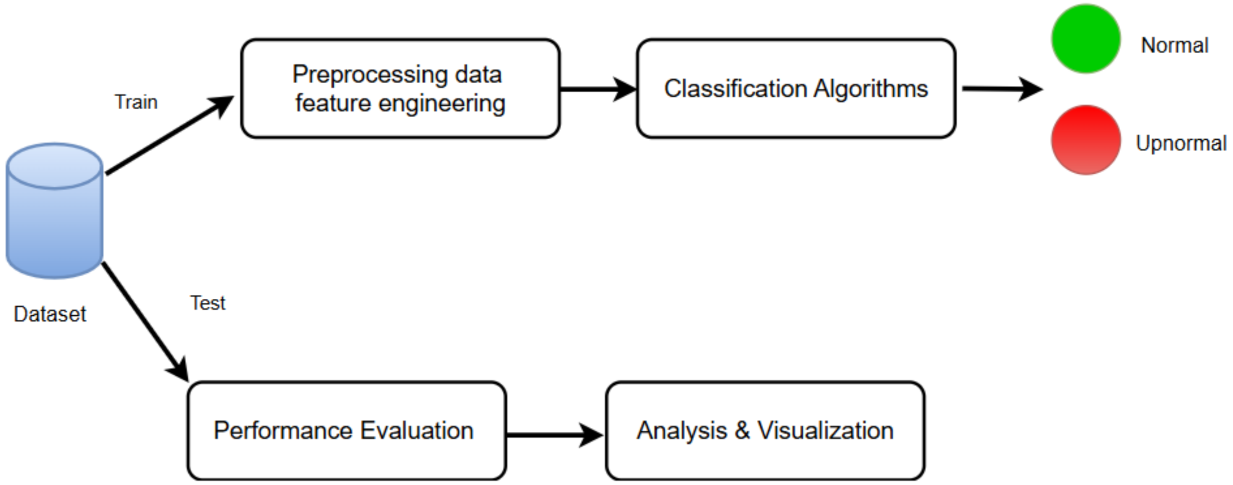


Figure.1. The proposed model

A. Artificial Neural network

Artificial Neural Networks (ANNs) are designed to emulate the complex network of neurons found in the human brain. Comprising a vast array of interconnected nodes, these networks operate collaboratively to process and interpret data, often producing useful outcomes. The architecture of an ANN is organized into several layers tasked with specific functions: the input layer, which captures all the initial data points for the model; the hidden layers, which are involved in the internal computation and analysis of the data; and the output layer, where the processed information is compiled and delivered as the final output [21].

The model consists of two layers with 64 and 32 neurons, followed by a softmax layer for classification. Dropout is used to reduce redundant information and computation load.

B. Random Forest

The Random Forest (RF) approach, a collective machine learning technique, is utilized for solving both classification and regression challenges. It functions by creating a multitude of decision trees during the training stage and produces the result by identifying the mode of the classes (in classification tasks) or by calculating the mean prediction (in regression tasks). The technique adopts a bagging strategy to boost the model's overall effectiveness, integrating extra randomness as the trees are being built. Instead of choosing the most important feature for a node split, it selects the optimal feature from a randomly chosen subset of features, thereby enhancing the model's precision through increased randomness [12].

The parameter tuning of RF is configured to 100 and 10 for the number of estimators and maximum tree depth, respectively, to avoid overfitting. To ensure robust splitting, each leaf node takes a minimum of 4 samples. A 5-fold cross-validation with GridSearchCV is performed to obtain the optimal parameters. No class weights or balanced class weights are employed to avoid bias due to class imbalance.

C. Support Vector Machine

Support Vector Machine (SVM) is a classification methodology developed by Corinna Cortes and Vladimir Vapnik. Serving as a classifier, it employs a specific learning algorithm to organize input data for classification purposes and is adaptable for use in regression scenarios as well. As a type of supervised learning, the SVM aims to differentiate between data points by maximizing the margin between the classes in an expanded dimensional setting. It achieves this by creating a hyperplane that distinctly divides the classes after the data has been transformed into a space of higher dimensions. Notably, SVM is proficient in executing non-linear classification effectively by leveraging the kernel trick [24].

An RBF kernel is employed in the SVM classifier with regularization parameter C ([1, 10]) and kernel coefficient gamma ([0.01, 0.1]). GridSearchCV optimized these parameters to ensure a robust model and maximize classification metrics.

D. Decision tree classifier

Decision trees can find broad applications in the areas related to machine learning, image processing, and pattern identification. They are flow-like models that integrate into one series of basic tests inside a smooth, well-organized model. Each test in this tree has a numerical feature against a threshold value. The process of formulating the set of principles guiding decision trees is usually easier than working out the numerical links between nodes in a neural network.

Decision trees are primarily used for classification, and hence, this is one of the most popular methods within Data Mining. Every tree is composed of nodes and branches. In any given category, for classification purposes, each node represents characteristics, while each branch reflects the possible value that could be taken by the node. Decision trees are especially renowned for being simple to comprehend and have promised robust performance over a broad spectrum of different types of data. For these reasons, decision trees have found their way into an immense array of applications [25].

Multiple values of max depth (None, 10, and 30) are assigned in the decision tree, with minimum samples split of 5 or 10. The tuning parameters were selected by 5-fold cross-validation to ensure a balance between classification metrics and avoiding bias.

IV. EXPERIMENT RESULTS

A. Data acquisition

This work utilizes the diabetes-prediction dataset from Kaggle [26] as a helpful tool to predict if a patient has diabetes. This dataset draws on both medicinal and demographic information and will classify people based on their diabetic condition. It includes, but is not limited to, data points like oldness, gender, Body Mass Index (BMI), high blood pressure status, occurrences of cardiovascular disease, history of tobacco use, HbA1c measurements, and glucose concentration, and are presented in Table II below.

The dataset is used to predict the presence of diabetes, using '1' and absence as '0'. It contains 9 columns, with a mix of integer, decimal, and textual-type data. There are 96,128 records in the dataset.

TABLE II. DATASET DESCRIPTION

No.	Attributes	Description
1	Gender	Identifies as male or female
2	Age	The individual's age in years
3	Hypertension	A condition often seen alongside diabetes, characterized by elevated blood pressure
4	Heart_disease	Includes various heart-related issues, such as coronary artery disease and heart failure
5	Smoking_history	Details regarding an individual's smoking habits
6	Bmi	Body Mass Index, an indicator of body fat derived from height and weight
7	Hba1c_level	Reflects the mean blood sugar concentration over the last 2-3 months

8	Blood_glucose_level	The concentration of sugar in the bloodstream at a specific moment
9	Diabetes	A binary indicator where 0 represents absence and 1 indicates presence of diabetes.

B. *Data pre-processing and feature engineering*

Preprocessing is one of the most important preparations for model training. Our dataset, comprising 96,128 rows and 9 columns (four integers, three decimals, and two strings), underwent comprehensive data cleaning and missing value handling. To effectively manage categorical features, we employed techniques in Feature Encoding, especially one-hot encoding, so that these variables are appropriately preprocessed by the machine learning algorithms. The distribution of the dataset is severely imbalanced, as shown in Figure 2; 91.5% of the samples belong to class 0 (normal class), while 8.5% belong to class 1 (diabetic class). Particularly in the artificial neural network algorithm we adopted here, the split of the data is 80% for the training process and 20% for testing.

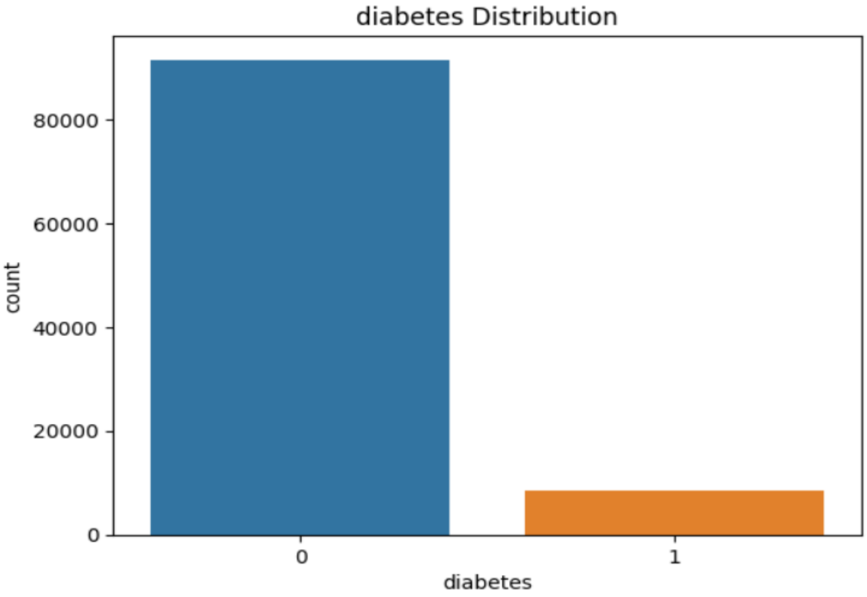


Figure.2. The distribution of diabet dataset

C. *Hyperparameters Tuning*

Following the preprocessing steps, we advance to the training of the models. We utilize GridSearchCV for the precise adjustment of hyperparameters in the random forest, decision tree, and SVM algorithms, as detailed in Table III. The refinement of hyperparameters plays a vital role in boosting the efficiency of machine learning models. It consists of identifying the ideal set of hyperparameters that prevent both overfitting and underfitting, thus guaranteeing the models operate at their highest capability.

TABLE III. THE BEST HYPERPARAMETER

Algorithm	**The best hyperparameter (using GridSearchCV)**
Random forest	{'class_weight': None, 'max_depth': 10, 'min_samples_leaf': 4, 'n_estimators': 100}
Decision tree	{'max_depth': 10, 'min_samples_split': 5}
SVM	{'C': 10, 'gamma': 0.1}

performance metrics

There are many factors to measure the performance of methods, which are explained below.

A. Confusion Matrix

The trained prototype is evaluated on the test set. A confusion matrix is used to visualize the model's performance [27]. The predictions of the model are shown in Figure 3. Where:

True Positive (TP): These are the cases where the actual class is positive (diabetes) and the model correctly predicted them as positive.
True Negative (TN): These are the cases where the actual class is negative (normal) and the model correctly predicted them as negative.
False Positive (FP): These are the cases where the actual class is negative (normal) and the model incorrectly predicted them as positive.
False Negative (FN): These are the cases where the actual class is positive (diabetes) and the model incorrectly predicted them as negative.

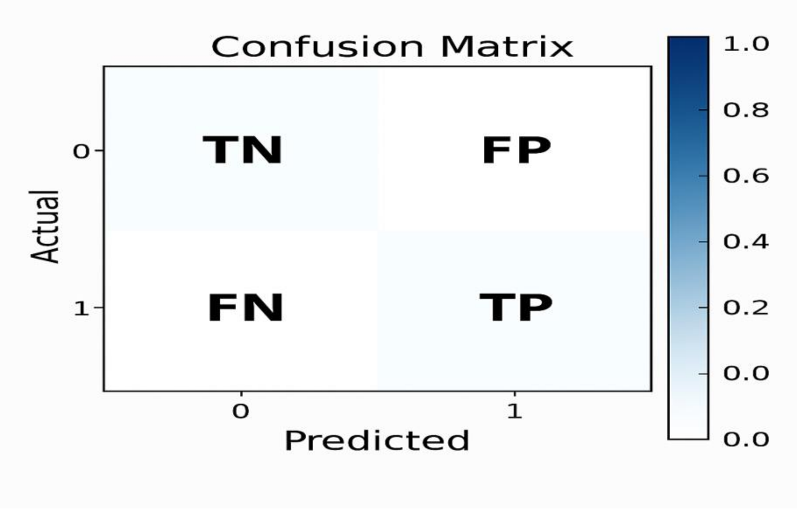


Figure. 3. Standard confusion matrix

B. Prescion

Precision (equation (1)) evaluates the correctness of positive identifications, defined by the proportion of true positives (TP) relative to all positive predictions made, which includes both true positives (TP) and false positives (FP).

Precision=TP/(TP+FP) □□□

C. Recall

Recall, or Sensitivity (equation (2)), assesses the ability to accurately detect real positive cases. It is determined by the ratio of true positives (TP) to the aggregate of actual positives, encompassing both true positives (TP) and false negatives (FN).

Recall=TP/(TP+FN) □□□

D. F1-score

The F1 score (equation (3)) serves as the harmonic mean of Precision and Recall, designed to equally weigh both metrics. It's computed by doubling the product of Precision and Recall, then dividing this by the addition of Precision and Recall.

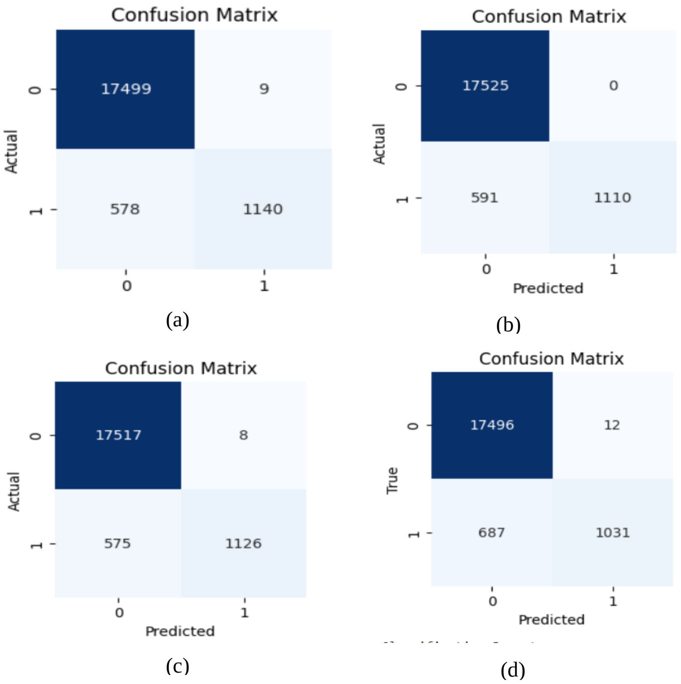
F1 Score=2∗Precision∗Recall/ (Precision + Recall□□□□□□□□□□□□□□□

In these formulas, True Positives (TP) are cases where a prediction of diabetes being present matches the actual condition. True Negatives (TN) are cases where both the prediction and actual condition agree on the absence of diabetes. False Positives (FP) describe scenarios where diabetes is incorrectly predicted to be present when it is not. Conversely, False Negatives (FN) are instances where diabetes is present but was not predicted.

V. THE RESULT AND DISCUSSION

The dataset was applied to the four algorithms, and the results gained from them are illustrated below.

The confusion matrix resulting from each algorithm is illustrated in Figure 4.



(a) (b) (c) (d)

Figure. 4. The resulted confusion matrix from (a) NN (b) RF (c) DT (d) SVM

There are minor variations across the results of the four algorithms. The Artificial Neural Network (ANN) records the greatest count of true positives (TP), which refers to accurately classifying abnormal patients. Meanwhile, the Random Forest (RF) algorithm achieves the highest true negatives (TN), which means it is most effective at classifying normal people.

For the FN, the artificial neural network and decision tree give nearly the same numbers as the decision tree, giving the lo+-west number, meaning it can reduce the number of patients with diabetes that are predicted as non-diabetes. The FP the random forest algorithm give the lowest number (zero) meaning it does not make a wrong prediction for the persons who are non-diabetes.

Finally, each model achieves good results in different areas, spotlighting trade-offs between minimizing FN and FP to ensure correct predictions.

TABLE IV. THE ACCURACY, PRECISION, RECALL, AND F1-SCORE

Algorithm	Tr Acc/ Ts Acc	Precision		recall		F1-score	
		0	1	0	1	0	1
NN	97 % 97.06%	0.97	0.99	1.00	0.66	0.98	0.80
RF	96.926%	0.97	1.00	1.00	0.65	0.98	0.79
DT	97.173% 96.967%	0.97	0.99	1.00	0.66	0.98	0.79
SVM	96.364%	0.96	0.99	1.00	0.60	0.98	0.75

Tr: Train, Acc: Accuracy, Ts: Test

From Table IV we can observe that the accuracy is nearly the same with no overfitting, while the artificial neural network have the highest accuracy. Despite the recall being relatively lower, this indicates that more improvement is required for positive case detection. Improvements such as feature selection and data augmentation can enhance positive case detection without compromising accuracy.

As for the F1 score, all algorithms yield a 98% score for class 0, but the neural network distinguishes itself with the highest F1 score of 80% for class 1. Regarding precision, the Random Forest algorithm outperforms others for class 1, showcasing its efficiency. For the recall of class, each algorithm demonstrates equal effectiveness, correctly classifying individuals as non-diabetic. The SVM achieves accuracy close to that of the ANN, but it achieves lower recall, which indicates it is critical to improve for use in the medical field. Accordingly, SVM needs other processes like data augmentation and feature selection to enhance its reliability for this task.

The accuracy curve of the Artificial Neural Network (ANN) is shown in Figure 5; the curve explains that the ANN model was effectively trained with minimal overfitting, as indicated by the close accuracy values between training and validation datasets.

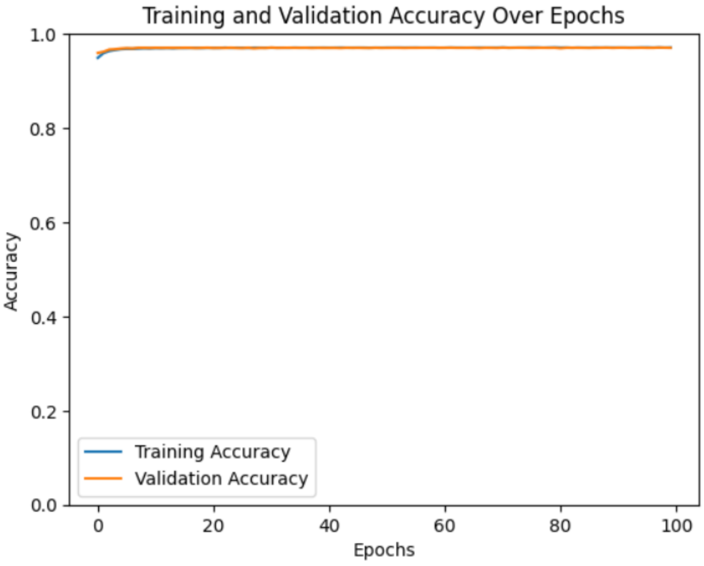
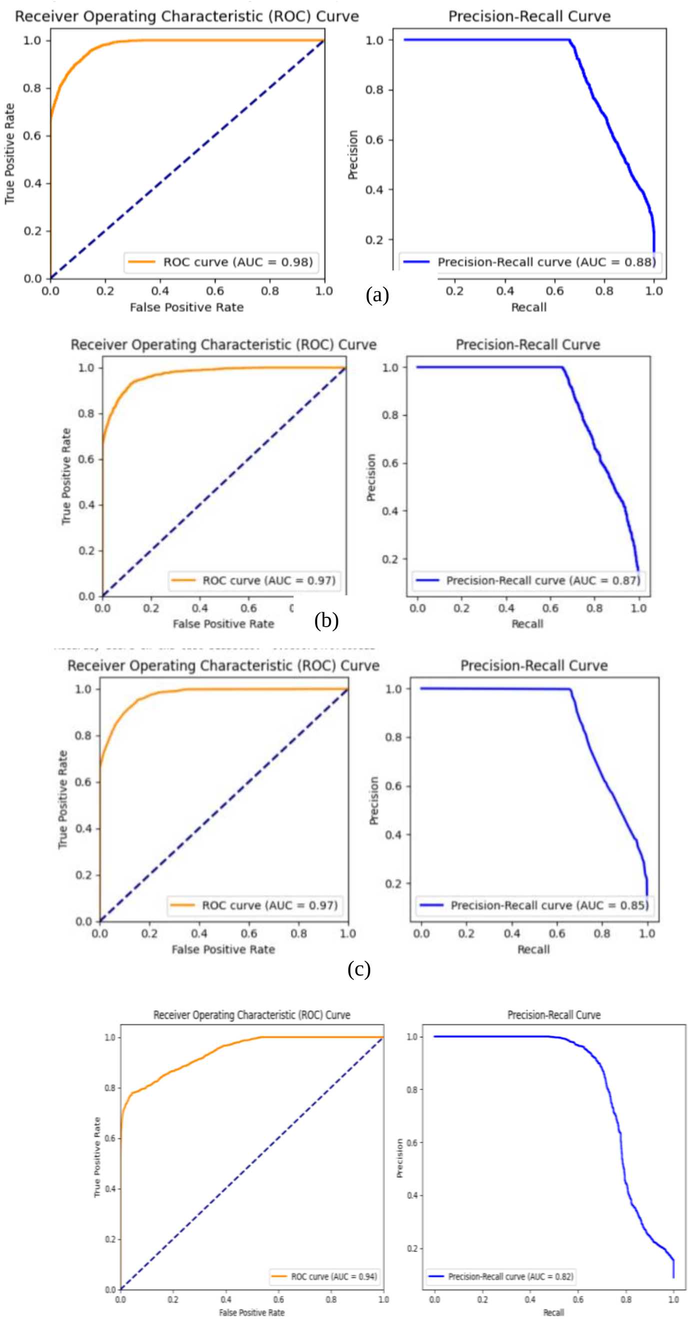


Figure. 5. The accuracy of NN

The ROC Curves help in understanding the balance between true positive rate and false positive rate at various threshold settings. PRC Curves are particularly useful in evaluating models on imbalanced datasets. They represent the equilibrium between precision and recall attained by modifying thresholds. A higher area under the curve (AUC) indicates both high recall and high precision, meaning that the classifier is returning more correct results as well as returning most of the all relevant results. The accuracy of a test increases as its ROC curve closely traces the left edge and subsequently rises along the top edge of the ROC space.



(a)

(b)

(c)

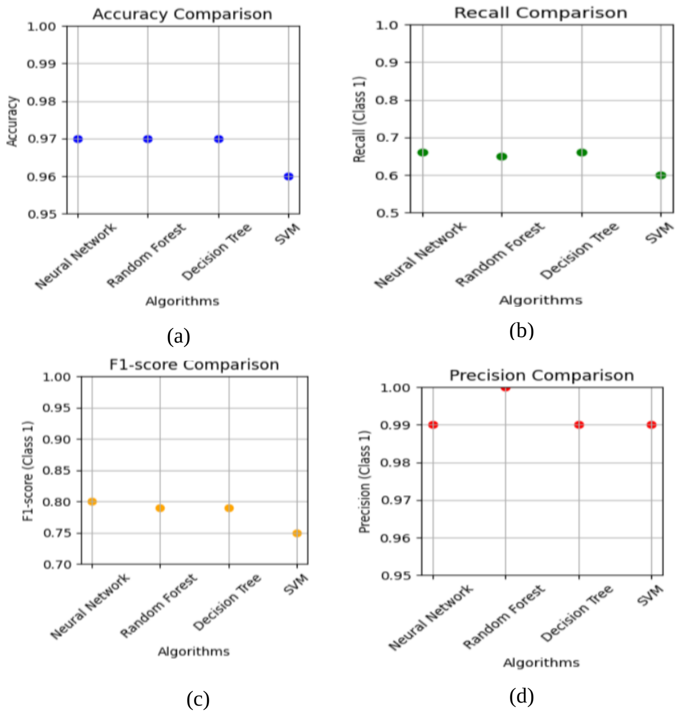
(d)

Figure. 6. PRC and ROC curves (a) ANN (b) RF (c) DT (d) SVM

Figure 6 shows the ROC and PRC curves for each method. The ANN has an AUC of 0.98 for ROC and 0.88 for Precision-Recall. Random Forest (RF) has ROC AUC: 0.97 and Precision-Recall AUC 0.87. DT has ROC AUC: 0.97 and Precision-Recall AUC: 0.85, finally Support Vector Machine (SVM) ROC AUC: 0.94 and Precision-Recall AUC: 0.82.

With the highest AUC in both the ROC and Precision-Recall curves, the ANN looks to perform the best among the four algorithms based on these AUC values. This implies that it offers the optimum compromise between precision and recall (Precision-Recall) and between sensitivity and specificity (ROC), two crucial criteria for classification tasks.

The ROC AUC analysis shows that both Random Forest and Decision Tree algorithms exhibit comparable performance, with Random Forest achieving a slightly higher Precision-Recall AUC. This could indicate that Random Forest performs slightly in an imbalanced datasets, particularly for identifying the positive class. Among the algorithms evaluated, SVM demonstrates the lowest AUC values for both curves, indicating a weaker performance, as well as the others, in distinguishing between the classes.



(a) (b)

(c) (d)

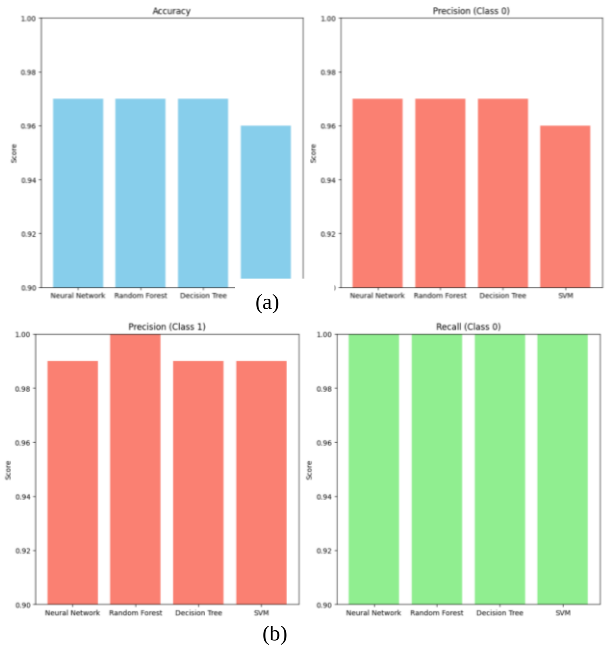
Figure. 7. Scatter plots

The scatter plots shed light on how various algorithms perform in relation to various performance parameters. Although the accuracy and precision of all algorithms are good, variations in recall and F1-score exist, suggesting variations in their capacity to accurately detect positive cases.

With accuracy values ranging from 0.96 to 0.97 for the Accuracy Comparison displayed in Figure 7(a), there is little variance in accuracy among the algorithms, demonstrating consistent performance across several models. The scatter plot in Figure 7(b)'s Recall Comparison illustrates each algorithm's recall score for class 1 (positive class). Algorithms' recall values range from 0.60 to 0.66, reflecting differences in their ability to reliably recognize positive cases. Compared to Random Forest and SVM, Artificial Neural Network and Decision Tree have greater recall ratings, indicating that they perform better at catching positive instances. The F1-scores, which represent the harmonic mean of precision and recall for the positive class, range from 0.75 to 0.80, as shown in the F1-score Comparison in Figure 7(c). Similar to the recall results, Artificial Neural Networks and Decision Trees achieve higher F1-scores than Random Forests and SVM, indicating superior overall performance in balancing precision and recall.

For the Precision Comparison shown in Figure 7(d), Precision values range from 0.99 to 1.00, reflecting the ability of each algorithm to correctly classify positive instances among all predicted positive instances. Every method exhibits a high degree of precision with very little fluctuation.

Contains bar charts for different performance metrics (Accuracy, Precision for Class 0 and Class 1, and Recall for Class 0) across several algorithms (Artificial Neural Network, Random Forest, Decision Tree, and SVM).



(a)

(b)

Figure 8. Charts for performance metrics across several algorithms

From Figure 8, the accuracy values for all models are again quite similar, indicating consistent overall performance. Regarding precision for Class 1, the SVM appears to have a slightly higher precision than the other models, suggesting it is marginally better at correctly identifying positive cases when predicting Class 1. In terms of precision for Class 0, the models perform nearly identically, with the Artificial Neural Network and SVM showing marginally better results. For recall in Class 0, all algorithms demonstrate almost identical and excellent performance, indicating their strong ability to identify negative cases correctly.

VI. CONCLUSION

Diabetes mellitus is a condition that can lead to numerous complications. This paper aims to explore four ML algorithms that might be used to accurately predict and diagnose this disease. The investigation reveals that the Artificial Neural Network (ANN) stands out by achieving a notable accuracy of 97%, while also effectively managing precision and recall for the minority class. Classification-based algorithms such as Random Forest and Decision Tree also demonstrate satisfactory performance, with only slight variations observed in their confusion matrices, highlighting their robustness in handling the dataset. In contrast, the Support Vector Machine (SVM) performs less effectively, with an accuracy of 96.36% and a lower recall for the minority class, indicating some challenges in addressing class imbalance.

ANN is an excellent algorithm that captures non-linear relationships within patient data, leading to accurate diagnosis of diabetes risk. Finally, these methods offer useful resources for risk assessment and early diagnosis of disease that help doctors manage diabetic health. Additionally, ANN, Random Forest, and Decision Tree maintain a good balance between precision and recall, which is a critical aspect for addressing imbalanced datasets in similar peer-reviewed studies.

This study illustrates the promise of different machine learning models in improving diabetes detection, which paves the way for future work in these models as they can be used together to enhance accuracy. It also stresses the need to improve how these models deal with imbalanced data for the purpose of building better diagnostic systems.

REFERENCES

[1] M. Maniruzzaman, M. J. Rahman, B. Ahammed, and M. M. Abedin, “Classification and prediction of diabetes disease using machine learning paradigm,” Heal. Inf. Sci. Syst., vol. 8, no. 1, 2020, doi: 10.1007/s13755-019-0095-z.

[2] S. Perveen, M. Shahbaz, A. Guergachi, and K. Keshavjee, “Performance Analysis of Data Mining Classification Techniques to Predict Diabetes,” Procedia Comput. Sci., vol. 82, pp. 115–121, 2016, doi: 10.1016/j.procs.2016.04.016.

[3] N. Nai-Arun and R. Moungmai, “Comparison of Classifiers for the Risk of Diabetes Prediction,” Procedia Comput. Sci., vol. 69, pp. 132–142, 2015, doi: 10.1016/j.procs.2015.10.014.

[4] J. P. Kandhasamy and S. Balamurali, “Performance analysis of classifier models to predict diabetes mellitus,” Procedia Comput.

Sci., vol. 47, pp. 45–51, 2015, doi: 10.1016/j.procs.2015.03.182.

[5] Á. C. Ribeiro, A. K. Barros, E. Santana, and J. C. Príncipe, "Diabetes classification using a redundancy reduction preprocessor," Rev. Bras. Eng. Biomed., vol. 31, no. 2, pp. 97–106, 2015, doi: 10.1590/1517-3151.0608.

[6] Y. Hayashi and S. Yukita, "Rule extraction using Recursive-Rule extraction algorithm with J48graft combined with sampling selection techniques for the diagnosis of type 2 diabetes mellitus in the Pima Indian dataset," Informatics Med. Unlocked, vol. 2, pp. 92–104, 2016, doi: 10.1016/j.imu.2016.02.001.

[7] F. Mercaldo, V. Nardone, and A. Santone, "Diabetes Mellitus Affected Patients Classification and Diagnosis through Machine Learning Techniques," Procedia Comput. Sci., vol. 112, pp. 2519–2528, 2017, doi: 10.1016/j.procs.2017.08.193.

[8] P. Indoria and Y. K. Rathore, "A Survey : Detection and Prediction of Diabetes Using Machine Learning Techniques," Int. J. Eng. Res. Technol., vol. 7, no. 03, pp. 287–291, 2018.

[9] M. Nilashi, O. Bin Ibrahim, A. Mardani, A. Ahani, and A. Jusoh, "A soft computing approach for diabetes disease classification," Health Informatics J., vol. 24, no. 4, pp. 379–393, 2018, doi: 10.1177/1460458216675500.

[10] R. B. Lukmanto, Suharjito, A. Nugroho, and H. Akbar, "Early detection of diabetes mellitus using feature selection and fuzzy support vector machine," Procedia Comput. Sci., vol. 157, pp. 46–54, 2019, doi: 10.1016/j.procs.2019.08.140.

[11] V. Rawat and Suryakant, "A classification system for diabetic patients with machine learning techniques," Int. J. Math. Eng. Manag. Sci., vol. 4, no. 3, pp. 729–744, 2019, doi: 10.33889/IJMEMS.2019.4.3-057.

[12] R. Deo and S. Panigrahi, "Performance Assessment of Machine Learning Based Models for Diabetes Prediction," in 2019 IEEE Healthcare Innovations and Point of Care Technologies, HI-POCT 2019, 2019, no. 11, pp. 147–150, doi: 10.1109/HI-POCT45284.2019.8962811.

[13] H. R. Divakar, D. Ramesh, and B. R. Prakash, "An Ontology Driven System to Predict Diabetes with Machine Learning Techniques," Int. J. Innov. Technol. Explor. Eng., vol. 9, no. 2, pp. 4005–4011, 2019, doi: 10.35940/ijitee.b7586.129219.

[14] J. Ye, L. Yao, J. Shen, R. Janarthanam, and Y. Luo, "Predicting mortality in critically ill patients with diabetes using machine learning and clinical notes," BMC Med. Inform. Decis. Mak., vol. 20, no. 11, pp. 1–8, 2020, doi: 10.1186/s12911-020-01318-4.

[15] G. L. A. Kumari, P. Padmaja, and J. S. G, "ENN-Ensemble based Neural Network Method for Diabetes Classification," Int. J. Eng. Adv. Technol., vol. 9, no. 3, pp. 574–579, 2020, doi: 10.35940/ijeat.c4819.029320.

[16] G. Mainenti, L. Campanile, F. Marulli, C. Ricciardi, and A. S. Valente, "Machine learning approaches for diabetes classification: Perspectives to artificial intelligence methods updating," in IoTBDS 2020 - Proceedings of the 5th International Conference on Internet of Things, Big Data and Security, 2020, pp. 533–540, doi: 10.5220/0009839405330540.

[17] Z. Bashir and H. Ghous, "A Comparative Analysis on Diabetes Datasets Using Data Mining Techniques," LC Int. J. STEM (ISSN 2708-7123), 2020.

[18] H. Zhou, R. Myrzashova, and R. Zheng, "Diabetes prediction model based on an enhanced deep neural network," Eurasip J. Wirel. Commun. Netw., vol. 2020, no. 1, 2020, doi: 10.1186/s13638-020-01765-7.

[19] R. H. Abiyev and H. Altiparmak, "Type-2 Fuzzy Neural System for Diagnosis of Diabetes," Math. Probl. Eng., vol. 2021, 2021, doi: 10.1155/2021/5854966.

[20] W. L. Alyoubi, M. F. Abulkhair, and W. M. Shalash, "Diabetic retinopathy fundus image classification and lesions localization system using deep learning," Sensors, vol. 21, no. 11, pp. 1–22, 2021, doi: 10.3390/s21113704.

[21] S. NAHZAT and M. YAĞANOĞLU, "Makine Öğrenimi Sınıflandırma Algoritmalarını Kullanarak Diyabet Tahmini," Eur. J. Sci. Technol., no. 24, pp. 53–59, 2021, doi: 10.31590/ejosat.899716.

[22] T. Dudkina, I. Meniailov, K. Bazilevych, S. Krivtsov, and A. Tkachenko, "Classification and prediction of diabetes disease using decision tree method," in CEUR Workshop Proceedings, 2021, vol. 2824, pp. 163–172.

[23] L. Fregoso-Aparicio, J. Noguez, L. Montesinos, and J. A. García-García, "Machine learning and deep learning predictive models for type 2 diabetes: a systematic review," Diabetology and Metabolic Syndrome, vol. 13, no. 1. BioMed Central, 2021, doi: 10.1186/s13098-021-00767-9.

[24] N. Izzati, A. Kader, U. K. Yusof, and S. Naim, "A Study of Diabetic Retinopathy Classification Using Support Vector Machine A Study of Diabetic Retinopathy Classification Using Support Vector Machine," 2021.

[25] A. M. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," 2021, doi: 10.38094/jastt20165.

[26] Mustafa T. (2021). Diabetes Prediction Dataset. Kaggle. Retrieved from https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset

[27] Abdalkareem, T. A., Zidan, K. A., & Albahri, A. S. (2024). A Systematic Review of Adversarial Machine Learning and Deep Learning Applications. Al-Iraqia Journal for Scientific Engineering Research, 3(4), 14-40.