Al-Kitab Journal for Pure Sciences (2026); 10(1):159-178.
https://doi.org/10.32441/kjps.10.1.p11

Al-Kitab Journal for Pure Sciences
ISSN: 2617-1260 (print), 2617-8141(online)

https://isnra.net/index.php/kjps

An Experimental Comparison of Different Machine Learning Algorithms for Detecting Type 2 Diabetes

Shnoo Abdul Aziz Zangana [1], Prof. Dr. Ahmad Hussain AlBayati [2], Farhan Naqee AlBayati [3]

[1,2] Department of Computer Science, Computer Science and Information Technology, University of Kirkuk, Iraq.

[3] Internal Medicine Department, Tikrit Teaching Hospital, University of Tikrit, Iraq.

*Corresponding Author: stcm22008@uokirkuk.edu.iq

***Citation**: **Shnoo Abdul Aziz Zangana [1], Asst. Prof. Dr. Ahmad Hussain AlBayati [2], Farhan Naqee AlBayati [3]** . **An Experimental Comparison of Different Machine Learning Algorithms for Detecting Type 2 Diabetes**. Al-Kitab J. Pure Sci. [Internet]. 2025 Jul. 22; 10(1):159-178. DOI:

https://doi.org/10.32441/kjps.10.1.p11.

Keywords: ML, Decision Tree, Naïve Bayes, Random Forest, Gradient Boosting, K-Nearest Neighbor, Logistic Regression & ANN.

Article History

Received	14 May. 2025
Accepted	22 Jul. 2025
Available online	01 May. 2026

Abstract:

Artificial intelligence (AI) is transforming healthcare, with (ML) offering significant potential for early disease diagnosis and personalized treatment, particularly in diabetes management. This capability is crucial for enhancing diagnostic accuracy. However, accurately predicting type 2 diabetes remains challenging due to diverse patient populations and complex data, as traditional methods can be slow and miss subtle early indicators, leading to delayed interventions. There is a clear need for robust predictive models. This paper proposes and evaluates various ML algorithms and data mining techniques for effective type 2 diabetes prediction. We utilized a unique Iraqi dataset with comprehensive features including HbA1c, lipid profiles, age, and BMI. Our methodology involved rigorous data preprocessing, including cleaning and handling class imbalance with SMOTE. Eight ML algorithms were compared: Decision Tree (DT), Naïve Bayes (NB), Random Forest (RF), Gradient Boosting (GB), K-Nearest Neighbor (KNN), Logistic Regression (LR), and Artificial Neural Networks (ANN). Model performance was assessed using accuracy, precision, recall, F1-score, MAE, RMSE, and AVE, with cross-validation ensuring robustness. The DT algorithm achieved the highest performance, with an accuracy of 99.44%, outperforming all other models. This highlights DT's

effectiveness and its potential for accurate early diagnosis, confirming HbA1c, age, and BMI as key predictors. Future work should establish a national health database in Iraq and explore advanced deep learning techniques, such as Convolutional Neural Network and Recurrent Neural Network.

Keywords: ML, Decision Tree, Naïve Bayes, Random Forest, Gradient Boosting, K Nearest Neighbor, Logistic Regression and ANN.

استخدام خوارزميات الذكاء الاصطناعي للتنبؤ بمرض السكري من النوع الثاني

شنو عبد العزيز صابر[1]، أ.م. د. أحمد حسين البياتي[2]، د. فرحان نقي البياتي[3]

اخصائي الباطنية /مستشفى تعليمي في تكريت/دائرة الصحة تكريت

stcm22008@uokirkuk.edu.iq,ahmad_taqi@uokirkuk.edu.iq,Tnftnf73@gmail.com

الخلاصة:

يُحدث الذكاء الاصطناعي (AI) تحولًا جذريًا في قطاع الرعاية الصحية، حيث يقدم التعلم الآلي (ML) إمكانات هائلة في التشخيص المبكر للأمراض والعلاج المخصص، لا سيما في إدارة مرض السكري. تُعد هذه القدرة حاسمة لتعزيز دقة التشخيص السريري. ومع ذلك، يظل التنبؤ الدقيق بمرض السكري من النوع الثاني تحديًا كبيرًا نظرًا لتنوع المجموعات السكانية للمرضى وتعقيد البيانات المتاحة، إذ قد تكون الأساليب التشخيصية التقليدية بطيئة وتفوت المؤشرات المبكرة الدقيقة، مما يؤدي إلى تأخر التدخلات العلاجية. لذا، هناك حاجة ماسة لتطوير نماذج تنبؤية قوية وموثوقة.

يُقدم هذا البحث وتقيّم مجموعة متنوعة من خوارزميات التعلم الآلي وتقنيات تنقيب البيانات بهدف التنبؤ الفعال بمرض السكري من النوع الثاني. لقد تم الاعتماد على مجموعة بيانات عراقية فريدة تتميز بخصائص شاملة، بما في ذلك الهيموغلوبين السكري (HbA1c)، وملفات الدهون (الكوليسترول، الدهون الثلاثية، HDL، LDL، VLDL)، والعمر، ومؤشر كتلة الجسم (BMI). تضمنت منهجيتنا معالجة مسبقة صارمة للبيانات، شملت التنظيف ومعالجة عدم توازن الفئات باستخدام تقنية SMOTE. تمت مقارنة أداء ثماني خوارزميات تعلم آلي: شجرة القرار (DT)، نايف بايز (NB)، الغابة العشوائية (RF)، تعزيز التدرج (GB)، أقرب جار (KNN)، الانحدار اللوجستي (LR)، والشبكات العصبية الاصطناعية (ANN). تم تقييم أداء النماذج باستخدام مقاييس الدقة، والضبط، والاستدعاء، ونتيجة F1، ومتوسط الخطأ المطلق (MAE)، ومتوسط الجذر التربيعي للخطأ (RMSE)، وتباين الأخطاء المطلقة (AVE)، مع ضمان متانة النتائج من خلال التحقق المتقاطع.

أظهرت خوارزمية DT أعلى أداء، حيث بلغت دقتها ٩٩,٤٤٪، متفوقة على جميع النماذج الأخرى التي تم اختبارها. يسلط هذا الإنجاز الضوء على فعالية خوارزمية DT وإمكاناتها في التشخيص المبكر الدقيق، مؤكدًا أن HbA1c والعمر ومؤشر كتلة الجسم هي المؤشرات الرئيسية للتنبؤ بالمرض. يجب أن تركز الأبحاث المستقبلية على إنشاء قاعدة بيانات صحية وطنية شاملة في العراق، مع إمكانية دمجها مع قواعد البيانات العالمية، واستكشاف تقنيات التعلم العميق المتقدمة، مثل الشبكة العصبية التلافيفية (CNN) والشبكة العصبية المتكررة (RNN)، لتعزيز القدرات التنبؤية.

الكلمات المفتاحية: التعلم الآلي، شجرة القرار، نايف بايز، الغابة العشوائية، تعزيز التدرج، أقرب جار K، الانحدار اللوجستي، الشبكات العصبية الاصطناعية.

1. Introduction

Artificial intelligence (AI) has become a rising star and is rapidly progressing, with the potential for incredible impact on healthcare (1). Artificial intelligence (AI) has created opportunities and transformed medical perspectives and diagnosis, and treatment paradigms. These advancements offer healthcare providers new ways of facing and solving formidable hurdles to expenditure reduction, disease management, service accessibility, and therapeutic optimization. Notable AI-based technologies, such as (ML) and deep learning (DL), have contributed significantly to diagnostics, patient monitoring, drug discovery, drug development, and telemedicine (2).

Diabetes is divided into three main types: type 1, type 2, and gestational diabetes. Type 1 diabetes is considered to be an autoimmune disease, in which the immune system attacks and destroys the cells in the pancreas that make insulin. Type 2 diabetes arises due to the inability of the body to make enough insulin and the failure of the body's cells to utilize insulin properly. Gestational diabetes, which occurs in pregnant women during the 6th and 9th months of pregnancy, is caused by the hormones the placenta makes, which in turn make it hard for the body's insulin to work. The proportion of subjects with type 2 diabetes is increasing versus those with type 1 and gestational diabetes.

Diabetes Mellitus (DM) is also known as diabetes, and is a long-term condition affecting the body's ability to convert food into energy (3). Algorithms are created using ML concepts to generate predictive models of the risk of DM onset and its complications. Applicable to resources-finite needs, the evidence of the efficacy of digital therapeutics has been established, as well as a validated intervention for lifestyle in the management of diabetes. Then, patients are increasingly empowered to self-manage diabetes, such that patients and clinicians are benefiting from clinical decision support (4). In this paper, methods for augmenting synthetic samples to under-represented classes were applied across various ML and DL techniques. These algorithms include Decision Tree (DT), Naïve Bayes (NB), Random Forest (RF), Gradient Boosting (GB), K-Nearest Neighbor (KNN), Logistic Regression (LR), and Artificial Neural Networks (ANN).

In this work, some pre-processing approaches have been employed for the performance enhancement of ML models. The dataset was first cleaned by dropping missing values, duplicated entries and outliers. Categorical attributes were converted to numbers using Label Encoding. To handle the class imbalance problem, the Synthetic Minority Over-sampling Technique (SMOTE) was applied to create synthetic instances for underrepresented classes. To

normalize the features and make the contribution of all features in learning equal, the Standard Scaler was used to scale the values of the features. Feature importance analysis, particularly with the DT model, also demonstrated that HbA1c, AGE, and BMI were the most significant predictors for diagnosing type 2 diabetes. Eight types of ML algorithms were developed: DT, NB, RF, GB, KNN, LR, and ANN. The performance of each model was evaluated using various metrics, including accuracy, recall, precision, AVE, MAE, F1-score, RMSE, and AUC.

Additionally, the cross-validation technique was applied to generalize and confirm the robustness of the results. This paper aims to develop accurate predictive models for identifying individuals at risk for type 2 diabetes at an early stage through ML based on a dataset with rich medical data. Using the most important predictors and by comparing models, the study is aimed at building a practical decision support for caregivers, especially in other countries with population demographics similar to Iraq.

2. Material and methods

2.1 Materials

The method is applied extensively in this study to the data at hand. In addition, the study relied upon the data of July 2020 obtained from Iraq, as provided by Information Technology University, which can be accessed through the link:

(https://data.mendeley.com/datasets/wj9rwkp9c2/1/files/2eb60cac-96b8-46ea-b971-6415e972afc9)

The diabetes data reported for the following 1000 records is categorized into the following sugar registers: (Yes) Y (diabetic), N (Non-Diabetic), P (Pre-Diabetic). It contains 12 features, which are: (Age, Gender, Cr, BMI, Urea, Chol (LDL, VLDL) Fasting Lipid Profile), TG, (HDL) High-density lipoprotein Cholesterol, and HbA1c. In this study, there were three categories: diabetic (Y), pre-diabetic (P), and non-diabetic (N). In order to reduce the classification problem to a binary one, we combined the person with diabetes (Y) and pre-diabetic (P) classes into a (positive) “diabetic” class, and the non-diabetic (N) class remained unaltered. This approach is in line with previous literature that stressed the clinical importance of identifying diabetic and pre-diabetic patients, as they have a higher chance of developing full diabetes and its complications (Edgar et al., 2024)(5). Thus, the binary classification separates the subjects at risk from those with a normal metabolism.

2.2 Methods

We have designed an effective classifier using ML for diabetes prediction. The code was written in Python and heavily uses the Pandas package for pre-processing the data, while the Label Encoder class is from the sklearn package. The preprocessing step involves converting the next part of the text features, which are currently categorical, to numeric as well.

2.2.1 Data Preprocessing

Preprocessing is the initial step to organize the input data for processing. It includes several processing operations, such as merge, reorder, and manipulation of data to clean, prune, reduce , and discretize (6). Choosing the right methods is key, as good data pre-processing can vastly improve classification results. To prepare the data for analysis, preprocessing involves several vital operations: eliminating noise (Noisy Data), Data Cleaning (Data Cleansing), Data Transformation, Data Reduction, and Data Integration. High-quality data must be guaranteed before analysis is conducted. These preparatory steps have previously been assumed by the author for high-quality data generation (7).

2.2.2 Data Cleaning

Data cleansing is the act of identifying and correcting (or removing) dirty data from a database. It usually involves discovering missing values, outliers, and inconsistencies, and then dealing with them by either excluding or substituting them with valid values (8).

2.2.3 Data Transformation

Data transformation is reformatting data for the analysis at hand. Data preprocessing is necessary to convert raw measurements into a format that can be analyzed by the analyzer, enabling the verification of the diabetes dataset.

2.2.4 Synthetic Minority Oversampling Technique (SMOTE)

SMOTE has been one of the most popular and efficient methods for addressing class imbalance problems in various fields. The basic intuition in SMOTE is to generate more minority samples that are close to some existing minority examples in feature space (9). To mitigate the class imbalance between the diabetic and the non-diabetic classes, we used SMOTE. The dataset was initially divided into the training and test sets to avoid data leakage. It was further used exclusively on the training part of the data to create synthetic samples for

the minority class. This process allowed for a consistent set of evaluation metrics that were valid and unbiased.

2.2.5 Data Splitting

The train_test_split method from the Scikit-learn library was utilized to separate the data into a training set (80%) and a test set (20%). This ensures that the models are trained on a varied training set and effectively evaluated on the test set. Moreover, cross-validation was applied to mitigate bias in evaluation as well as guarantee the performance of the model on other datasets. Specifically, the data is split into five groups (five-folds). The model is trained on four of these groups and tested on the remaining group, and this process is repeated for each group.

2.2.6 Utilized ML algorithms

a. Decision Tree (DT)

The DT algorithm is a supervised ML technique that divides the data in terms of some parameter. DT is a technique to separate a dataset into smaller sets by applying rules. That is, it may be described as a method that classifies (large) amounts of data into small groups of data **(10)**.

b. Naïve Bayes (NB)

The Bayesian classifier is a statistical classifier that uses conditional probability to categorize data into predefined categories. Both a descriptive and a predictive algorithm are considered to be NB. The categories of the untrained data are then predicted using the descriptive probability. The following are some advantages of this approach. It is simple to use to start. Furthermore, NB doesn't always require a lot of training data for categorization **(11)**.

c. Random Forest (RF)

The RF is an ensemble algorithm, which is capable of tackling both classification and regression tasks. RF is a classifier that includes multiple DT classifiers, with the results determined by majority voting from the metaclassifier. Each tree has been trained with a random sampling of the entire training set. Unlike some other algorithms, in the RF procedure, there is no problem with fitting the data too much (overfitting). The rate at which it fails can be used to compute the generalization error directly without the use of

a cross-validation procedure. The RF model can handle both categorical and numeric data, including missing values and non-scaled data (9).

d. Gradient Boosting (GB)

GB is an ML algorithm used for regression and classification. It is a type of ensemble learning method that combines weak learners to form a strong one. Successively fitting decision trees reduces the loss function to the residuals of the current model. Each iteration will focus more on the mistakes made in the last iteration. The output of all trees included in the model is combined to yield the final prediction (12).

e. K Nearest Neighbor Algorithm (KNN)

KNN is a minimal and popular modeling technique. Classification problems, pattern recognition and regression through ML algorithms. KNN gets neighbors from data by using the Euclidean distance between points of data (13).

f. Logistic Regression (LR)

LR is a supervised learning model commonly used to solve the binary classification problem. It applies the logistic function to estimate the probability of the data point being assigned to one of the categories. There are many complicated generalizations of LR. It is a model of regression; it predicts the probability of an item being a certain type (13).

g. Artificial neural networks (ANN)

ANN is an intelligent method to diagnose and predict diseases. ANN is an ML methodology that imitates humans to learn and predict intricate issues with neural architecture (14). A neural network is trained by providing it with patterns and allowing it to modify its weights in accordance with a given learning rule. The operation of the artificial neural network is divided into two stages: learning and inference. Before a decision is taken, the ANN is educated on previous examples of evaluation. Learning takes place after continually adjusting the weights until the time-average mean square error reaches the given minimum allowed threshold. After training, the decision-making remains in the weights and connections, as the trained model is now ready to solve a new case (15). Two types of ANN are presented as shown below.

A. Hidden Layer Perceptron Neural Networks (HLNN)

A neural network model containing a hidden layer is proposed and implemented to enhance the predictive ability of the model. This neural network has a very simple network structure consisting of an input layer and an output layer. It is a Perceptron that uses Back propagation to learn weights from the data in an efficient manner. Forward propagation generates predictions and calculates the error between the actual and predicted value so that the weights can be adjusted accordingly. The method ensures continuous improvement and is thus a useful tool for data-driven operations.

B. Multilayer Perceptron Neural Networks (MPNN)

A neural network model with multiple hidden layers is proposed and implemented to capture the complex relationships between input features. It is implemented to enhance the model's performance. The Multilayer network consists of a dynamic system. The input layer has nodes equal to the number of characteristics, and the hidden layer contains 200 nodes. Every node multiplies its inputs with its weights and passes them through a sigmoid activation function. The output layer of the present model contains a single node, which takes data from the hidden layer as input and applies the sigmoid function to map the output between 0 and 1.

3. Model Evaluation

Several evaluation measurements evaluate the predicted results. The performance is measured by classification accuracy, confusion matrix, and F1-score. The classification accuracy is defined as the number of correctly matched ones over the total number of input patterns. The applied metrics are as given below:

1. Root mean square error (RMSE): It is a very popular way to check the accuracy of the model for predicting statistical data. The Root Mean Square Error (RMSE) scores are between 0.0 and 0.5, indicating that the model is able to predict the data correctly. It is formulated as follows (16):

$$\sqrt{R}MSE = \frac{1}{m}\sum_{i=0}^{m}(xi - yi)^2 \qquad (1)$$

2. Mean absolute error (MAE): The mean absolute difference between the actual output and the predicted output for the entire dataset (16). MAE can be expressed as follows (17):

$$MAE = \frac{1}{m}\sum_{i=0}^{m} |xi - yi| \quad (2)$$

3. Kappa statistic: The kappa statistic quantifies the degree of concordance of the instances being classified by the ML classifier and conditions on the true labelled data as a ground truth to account for the expected accuracy of a random classifier. To see how well a classifier works for a given dataset, the following equation presents the kappa computation (16):

$$Kappa = \frac{OA - i}{1 - IE} \quad (3)$$

4. Accuracy is the proportion of correctly predicted positives, both true and false, out of all positives. It is composed as follows (18):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

5. Precision: is the ratio of the number of true positive predictions to the total number of positive predictions made by the model. It is calculated as follows (18):

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

6. Recall: It can be computed as the ratio of correctly estimated data points out of the total included true data points. Sensitivity is defined as follows (19):

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

7. F1-score: The F1-score is the harmonic mean of sensitivity and precision, and is determined as follows (18).

$$F1\_Score = 2\ X\ \frac{Precision * Recall}{Precision + Recal} \quad (7)$$

8. Confusion Matrix: A confusion matrix describes the predicted and actual classifications. The performance of any system is usually tested on matrix-shaped data (20).

9. ANN Evaluation: Here, we address research questions regarding the evaluation of ANN models.

RQ1: How to tune the Batch size and number of epochs?

RQ2: How to tune the training optimization algorithm?

RQ3: How to tune learning rate and momentum?

RQ4: How to tune network weight initialization?

4. Results

According to several metrics, including Accuracy, Recall, F1-score, Precision, Cohen's Kappa, MAE, RMSE, and running time. The DT model had the best predictive performance compared to the rest of the models. It reached the highest values of accuracy (99.4%), recall (98.6%), precision (98.64%), and F1-score (98.60%), as well as the best Cohen´s Kappa value (0.988), which denotes a very strong agreement with the reference labels. It was also the least error-prone (MAE = 0.005, RMSE = 0.07) and fastest in execution (13 ms), making it accurate and computationally efficient. The RF model had the same accuracy (99.4%) but lower values on the other measures. GB and KNN also achieved higher accuracies of 99.1% and 98.9%, respectively, but with higher residual errors. LR and NB achieved not bad results, obtaining, respectively, 97.8 and 97.9% accuracy. Among the presented ANN models, HLNN has the highest performance (97% accuracy), lowest error rate, and fastest processing time (0.4 ms). In comparison, MPNN has the lowest performance (95% accuracy) and has the longest processing time (64 seconds) compared to HLNN. Results from feature importance analysis also showed that HbA1c, AGE and BMI were the most important features for predicting type 2 diabetes. These features, in conjunction with the DT classifier, produced the highest accuracy (99.44%).

4.1 Tuning the Batch Size and Number of Epochs

The value of Batch Size is calculated, which is the number of training records passed to the model, i.e., the number of samples the model will treat. Even if the batch size is 20, the model will consider this value and calculate the error, then update the weights for each epoch. The best performance of 67% has been achieved precisely for payment size 20 and number of application training 100. The final result of this method is shown in Figure 1:

```
Best: 0.674040 using ('batch_size': 20, 'epochs': 100)
0.632317 (0.050247) with: ('batch_size': 10, 'epochs': 10)
0.650541 (0.029356) with: ('batch_size': 10, 'epochs': 50)
0.668903 (0.042884) with: ('batch_size': 10, 'epochs': 10)}
0.619373 (0.043666) with: ('batch_size': 20, 'epochs': 10)
0.646620 (0.039530) with: ('batch_size': 20, 'epochs': 50)
0.674040 (0.029795) with: ('batch_size': 20, 'epochs': 100)
0.611392 (0.057363) with: ('batch_size': 40, 'epochs': 10) .....
Accuracy: 61%
```

Figure 1: Network results for batch size and iteration count searches.

4.2 Tuning the Training Optimization Algorithms

It is trained with the aid of Kera's Library using a neural network. The search for this test is then employed to test various modifications or algorithms, adopting the one that achieves the best results. The current network is prepared using (EPOCH = 100) and the number of samples in each training step (Batch Size = 10), and the name of algorithms for optimization, which (SGD, RMSPROP, Adagrad, Adadelta, Adam, Adamax, Nadam) are used. The data is partitioned into three groups (Cross Validation). The resulting NADAM optimization algorithm was extracted. The best performance for the remaining optimizations was 67% (see Figure 2).

```
Best: 0.689711 using ('optimizer': 'Nadam')
0.653196 (0.001206) with: ('optimizer': 'SGD')
0.649290 (0.011847) with: ('optimizer': 'RMSprop')
0.585565 (0.100332) with: ('optimizer': 'Adagrad')
0.620639 (0.043876) with: ('optimizer': 'Adadelta')
0.688399 (0.009652) with: ('optimizer': 'Adam')
0.654539 (0.025063) with: ('optimizer': 'Adamax')
0.689711 (0.015514) with: ('optimizer': 'Nadam')
...
Accuracy: 0.7561517429938484
```

Figure 2: Network results for search optimization algorithms

4.3 Tuning Learning Rate and Momentum

This approach implements a neural network using TensorFlow/Keras, and additionally, the network search is used to obtain the best momentum and learning rate. The first step is to create a Model() and call the sequential function within it. This function is part of the Kiras model from the serial layers, allowing you to obtain the model. In the last layer, a person's binarized classification is determined, whether they have sugar or not. The Sigmoid activation function is used in the final layer to fit the value between (0,1). The form is warped using Kera's Classifier to determine the optimal parameters. Then, the learning and momentum rates are adjusted, and several values are tested to identify the best one. The values of learning rate used are (Learn rate = [0.001, 0.01, 0.1, 0.2, 0.3]), where the best value of the learning rate is 0.1. For momentum, several values were tested, and the following were experimented with: Momentum [0.0, 0.2, 0.4, 0.6, 0.8, 0.9]. The model performed best when congestion was

moderate (0.8). The results obtained using (SGD) with learning rate (0.8) and momentum (0.1) for this approach are: The best results are obtained with 75% accuracy (see Figure 3).

```
Best: 0.769982 using ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.8)
0.725024 (0.014011) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.0)
0.723762 (0.016650) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.2)
0.746299 (0.035143) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.4)
0.746271 (0.021892) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.6)
0.747520 (0.017654) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.8)
0.751298 (0.029039) with: ('optimizer__learning_rate': 0.001, 'optimizer__momentum': 0.9)
0.746243 (0.007302) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.0)
0.746271 (0.019400) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.2)
0.750035 (0.028135) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.4)
0.752513 (0.018427) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.6)
0.733759 (0.005724) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.8)
0.755005 (0.006860) with: ('optimizer__learning_rate': 0.01, 'optimizer__momentum': 0.9)
0.756235 (0.011355) with: ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.0)
0.758746 (0.002197) with: ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.2)
0.732454 (0.028245) with: ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.4)
0.745023 (0.013619) with: ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.6)
0.769982 (0.014745) with: ('optimizer__learning_rate': 0.1, 'optimizer__momentum': 0.8)
```

Figure 3: Network results for momentum and learning rate searches

4.4 Tuning Learning Weight Initialization

In this work, a neural network is developed using Keras 46 and features a hidden layer with 12 neurons. In this layer, ReLU is also used. The output layer is one cell using Sigmoid, and the model is optimized by GridSearch function Cv. It's a method for finding the best values for the hyperparameters of the model by iterating through all the possible sets of values. Then there were several methods which are: ('uniform', 'lecun_uniform', 'normal', 'zero', 'glorot_normal', glorot_uniform ', 'he_normal' ', 'he_uniform') and any of them that do better.

The data is divided into 10 parts, and cross-validation is used, with one part reserved for testing and the remaining nine parts for training the model. Between these types, the 'normal' type

gives the best performance compared with the rest, and then predicts (71% accuracy) and the generator output of this method, as mentioned in Figure 4.

```
Best: 0.718413 using ('model__init_mode': 'normal')
0.707945 (0.013441) with: ('model__init_mode': 'uniform')
0.689726 (0.021351) with: ('model__init_mode': 'lecun_uniform')
0.718413 (0.020539) with: ('model__init_mode': 'normal')
0.651889 (0.000643) with: ('model__init_mode': 'zero')
0.717075 (0.012976) with: ('model__init_mode': 'glorot_normal')
0.704013 (0.019934) with: ('model__init_mode': 'glorot_uniform')
0.685784 (0.006825) with: ('model__init_mode': 'he_normal')
0.683221 (0.025060) with: ('model__init_mode': 'he_uniform')
```

Figure 4: Network results for network weight configuration searches

5. Discussion

The current data set is obtained from Mendeley and has been properly synchronized to remove missing and outlier values. Categorical values are converted to numerical values, and duplicates are eliminated. Additionally, the technology of budget categories is used to increase the artificial samples of the less common category. Cross-validation is adopted, a statistical technique that splits data into a training dataset and a test dataset, assessing the model's compatibility and stability to be more accurate. From the evaluation standpoint, the accuracy, Precision, recall, F1 score, average absolute error, root mean square error, and average variance extracted (AVE) are used, as seen in Tables 1 and 2. An extensive exploration and simulation of machine learning algorithms are conducted to precisely diagnose diabetes at an early stage in the healthcare industry. A few ML techniques are selected, including RF, DT, GB, NB, KNN, LR, and ANN, Figure 9 illustrates the accuracy comparison among classification algorithms, showing that Decision Tree and Random Forest achieved the highest accuracy. With the current discovery, there is potential to change the computation of predicting and treating Diabetes, and ultimately enhance patient wellness . TP, or true positive, is the number of cases correctly classified as sick, indicating how well the current classifier is performing. (TN), A true negative is a case identified as having the disease, but it actually doesn't have the disease. False positives (FP) are cases where healthy conditions are incorrectly identified as unwell, and false negatives (FN) are cases where sick conditions are incorrectly considered healthy.

DT achieved the best overall performance across all studied algorithms and evaluation metrics. It obtained an accuracy of 99.4%, a recall and F1-score of 98.6%, and the best Cohen's Kappa value (0.988), surpassing the ensemble models, such as RF and GB. Confirming that DT was well adapted to this dataset, as it was able to handle mixed data types, model non-linear relationships, and provide rules that can be exhibited. RF and GB provided strong results as well, but with more complex models, they did not offer much improvement over DT, which performed just as well and faster. Performance results of neural network models varied the most over the compared methods. MPNN showed competitive F1 and precision, but is not suitable for real-time scenarios due to its very high runtime (64 s). In comparison, HLNN achieved a better trade-off between speed and accuracy, making it more suitable for fast diagnosis.

Future research should focus on: Establishment of a national database encompassing all health centers and clinics in Iraq, with the capability for integration with worldwide or neighboring databases, Creating a consolidated center between the Ministry of Health and the Ministry of Higher Education, It is recommended to utilize a CNN and RNN model that employs cutting-edge deep learning techniques.

Feature importance measures the significance of each feature in the decision-making process of the data tree. It is a value between 0 and 1 for each feature, where zero indicates "never used" and 1 indicates "accurate prediction of the target" (see Figure 5).

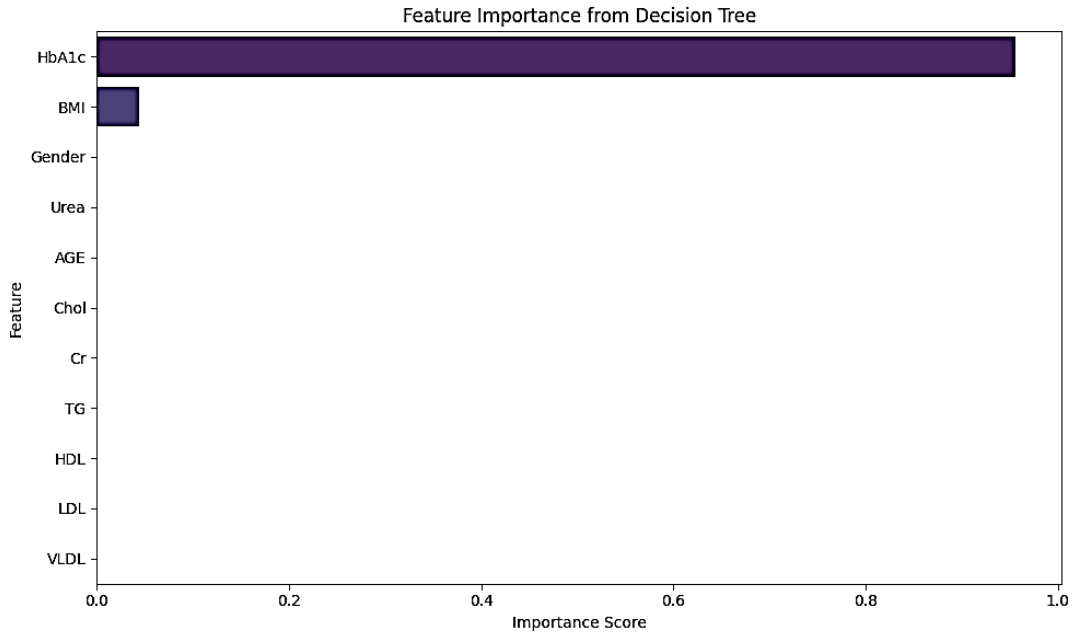

Figure 5: Feature importance in the DT model

The two most important features in the data tree are glycated hemoglobin (HbA1c) and body mass index (BMI). The major predictor for diabetes was HbA1c, which was a full and nearly perfect predictor, followed by BMI, which had a small effect. The other characteristics had virtually no additional effect, and HbA1c alone was sufficient to separate the samples correctly.

According to Figure 6, all the features in the LR model are listed from highest to lowest importance. The data are HbA1c, BMI, Cholesterol, TG, etc.

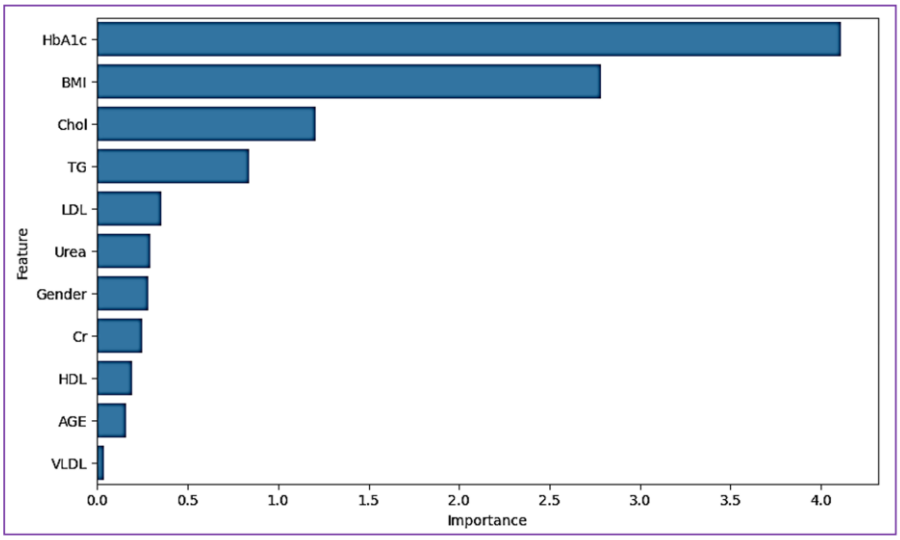

Figure 6: Feature importance in the LR model

The efficiency of the trained classifier model (RF classifier) was assessed using the error matrix (Confusion Matrix). This matrix provides an overview of the model's classification of each sample, as well as the correct and incorrect classifications for each class. The error matrix gathered here will be described and discussed. Figure 7 illustrates the confusion matrix, where the classification results of the RF algorithm on the binary diabetes data, where there are two classes: Class 0 (non-diabetic) and Class 1 (Diabetic).

The confusion matrix demonstrates the following measurements. Firstly, TN = 179, which represents the number of samples that actually belong to the uninfected class and were correctly classified. Secondly, FP = 0, which means no mistake is committed in classifying uninfected samples as infected. This is particularly good since we didn't have any false alarms. Thirdly, FN = 2, where two cases were incorrectly not identified as infected, despite being infected. Sensitivity to this kind of error is important in a medical use case. Lastly, TP = 178, representing the number of infected samples classified correctly. This figure illustrates the model's close performance in diagnosing infected individuals.

The confusion matrix, the overall accuracy index of the model, as well as more precise performance indicators, including precision, recall, and F1 score, were evaluated using the test set. These evaluations were further enhanced by using a 5-fold cross-validation to obtain a more comprehensive and stable evaluation of model performance.

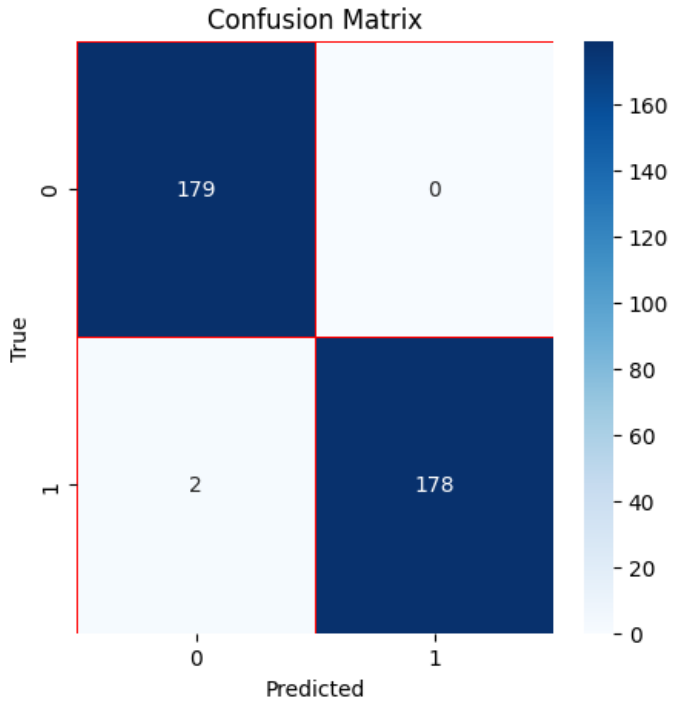

Figure 7: Confusion matrix of the RF model

This was supported by the high scores obtained in ROC AUC, with most models achieving 100% using this metric, such as RF, GB, and LR models. The DT, KNN and NB models also achieved a score of 99%. These findings indicate the discriminative value of the models in distinguishing patients with T2DM from those without T2DM. They also demonstrate that the models are utilizing more effective features in the data, beyond just superficial accuracy, which thus helps to boost their reliability in medical diagnostic scenarios. Figure 8 illustrates a cross-comparison of ROC curves in various models.

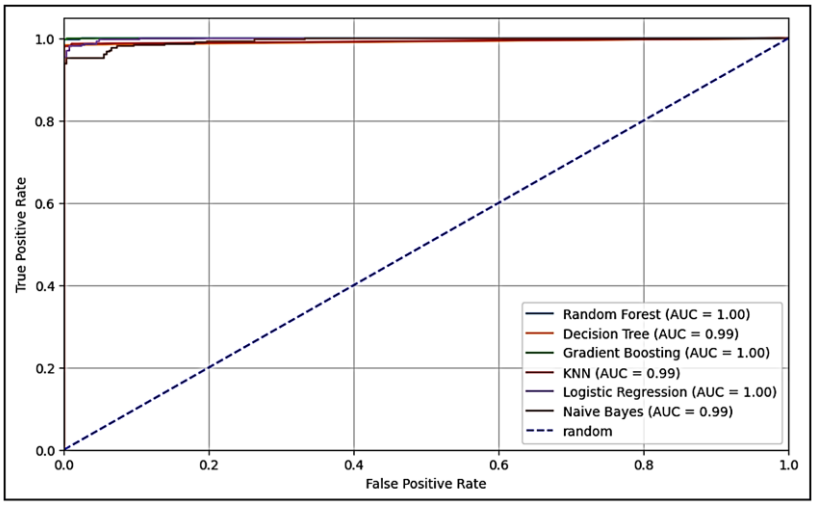

Figure 8: ROC curve of all models developed using the IRD dataset

Table 1: Performance evaluation metrics for the diabetes dataset

NO.	*Algorithm*	*Accuracies*	*Recall*	*F_Score*	*Precision*	*Cohen kappa*
1.	R F	0.994	0.983	0.983	0.984	0.988
2.	DT	0.994	0.9860	0.9860	0.9864	0.988
3.	GB	0.991	0.985	0.985	0.9863	0.983
4.	NB	0.979	0.9667	0.9666	0.968	0.959
5.	KNN	0.989	0.970	0.969	0.971	0.979
6.	LR	0.9781	0.970	0.969	0.971	0.956
7.	MPNN	0.9500	0.9609	0.9718	0.9829	0.7546
8.	HLNN	0.9700	0.9721	0.9831	0.9943	0.8528

Table 2: The error rate analysis

NO.	*Algorithm*	*MAE*	*RMSE*	*AVE*	*Time*
1.	R F	0.005	0.07	0.005	34ms
2.	DT	0.005	0.07	0.005	13ms
3.	GB	0.008	0.09	0.008	5.82ms
4.	NB	0.02	0.14	0.01	1.59ms
5.	KNN	0.010	0.10	0.010	11.9ms
6.	LR	0.02	0.14	0.01	0.54ms
7.	MPNN	0.05	0.22	0.04	64150ms
8.	HLNN	0.03	0.17	0.02	0.4ms

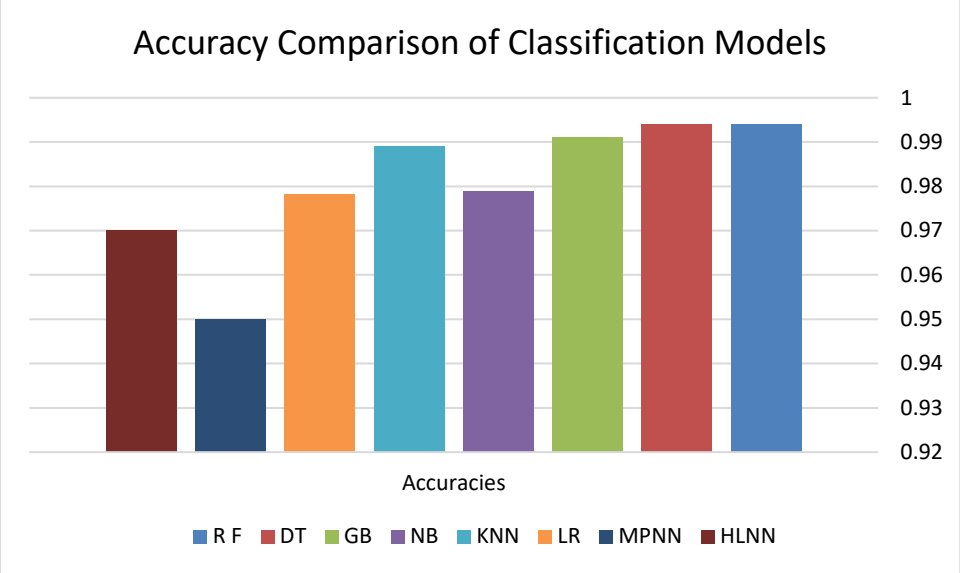

Figure 9: Comparison of the accuracy of algorithms

6. Conclusions

Diabetes is one of the most prevalent and highly prevalent chronic diseases worldwide. DM is a significant global health concern, with its prevalence rapidly increasing. Undiagnosed diabetes can lead to numerous complications, including retinopathy, nephropathy, neuropathy, and various vascular disorders. Both type 1 and type 2 diabetes are among the leading causes of mortality worldwide and are associated with renal disease, visual impairment, and cardiovascular conditions. Data mining techniques can enhance healthcare decision-making by facilitating accurate disease diagnosis and treatment, thereby reducing the burden on healthcare specialists. Using clinical data from Iraq, this study sought to develop and assess artificial intelligence models for the prediction of type 2 diabetes mellitus (T2DM).

The primary contribution is that the DT algorithm was shown to be the most effective model, outperforming all other tested models with a classification accuracy of 99.44%. The results emphasize the importance of features and data quality in improving model performance, particularly for biomarkers such as HbA1c and triglycerides. The results indicate that valuable clinical data is more important for diagnostic success than algorithm complexity. It also indicates the potential for investigating the construction of prototype models for intelligent medical electronic apparatus for diagnosing early diseases. The DT model is a strong candidate for integration into clinical decision support systems (CDSS), which are intelligent platforms that help medical professionals diagnose illnesses, provide treatments, and make well-informed decisions. This is due to the model's high accuracy and interpretability. Early diabetes risk

detection is made possible by integrating the suggested model into these systems, particularly in primary care and resource-constrained environments.

To improve generalizability and facilitate widespread clinical deployment, future research should concentrate on integrating real-time data streams from wearable devices, implementing longitudinal monitoring, and validating the models across various clinical populations.

7. References

1. Ziajor S, Tomasik J, Sajdak P, Turski M, Bednarski A, Stodolak M, Szydłowski Ł, Żurowska K, Krużel A, Kłos K, Dębik M. The use of artificial intelligence in the diagnosis and detection of complications of diabetes. Journal of Education, Health and Sport. 2024 Apr 11;65:11-27.
2. Iqbal J, Jaimes DC, Makineni P, Subramani S, Hemaida S, Thugu TR, Butt AN, Sikto JT, Kaur P, Lak MA, Augustine M. Reimagining healthcare: unleashing the power of artificial intelligence in medicine. Cureus. 2023 Sep 4;15(9).
3. Ismail L, Materwala H, Tayefi M, Ngo P, Karduck AP. Type 2 diabetes with artificial intelligence machine learning: methods and evaluation. Archives of Computational Methods in Engineering. 2022 Jan;29(1):313-33.
4. Ellahham S. Artificial intelligence: the future for diabetes care. The American journal of medicine. 2020 Aug 1;133(8):895-900.

Ceh-Varela E, Maes L, Shakya SR. Machine learning analysis of factors contributing to Diabetes Development. Cloud Computing and Data Science. 2024 Jan 3:157-82.

6. Ahmad Hussain AlBayati SAAZ. The Role of ML Algorithms for Diagnosing Diabetes Mellitus Based on Different Datasets with Different Attributes Journal of Information Systems Engineering and Management 2025; {10}.
7. Febrian ME, Ferdinan FX, Sendani GP, Suryanigrum KM, Yunanda R. Diabetes prediction using supervised machine learning. Procedia Computer Science. 2023 Jan 1;216:21-30.
8. Mohammed EM, Fakhrudeen AM, Alani OY. Detection of Alzheimer's disease using deep learning models: A systematic literature review. Informatics in Medicine Unlocked. 2024 Jan 1;50:101551.
9. Baker MR, Mahmood ZN, Shaker EH. Ensemble Learning with Supervised Machine Learning Models to Predict Credit Card Fraud Transactions. Revue d'Intelligence Artificielle. 2022 Aug 1;36(4).
10. Akmeşe ÖF. Diagnosing Diabetes with Machine Learning Techniques. Hittite Journal of Science and Engineering. 2022;9(1):9-18.
11. Chang V, Bailey J, Xu QA, Sun Z. Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Computing and Applications. 2023 Aug;35(22):16157-73.

12. Ahmed M, Husien I. Heart disease prediction using hybrid machine learning: A brief review. Journal of Robotics and Control (JRC). 2024 May 3;5(3):884-92..
13. Ibrahim I, Abdulazeez A. The role of machine learning algorithms for diagnosing diseases. Journal of Applied Science and Technology Trends. 2021 Mar 19;2(01):10-9.
14. Thompson C, Higgins O. Combination of Artificial Neural Network and Particle Swarm Intelligence Algorithm for Diagnosing Diabetes. Advances in Engineering and Intelligence Systems. 2024 Mar 30;3(01):23-33.
15. Ogwueleka FN, Misra S, Colomo-Palacios R, Fernandez L. Neural network and classification approach in identifying customer behavior in the banking sector: A case study of an international bank. Human factors and ergonomics in manufacturing & service industries. 2015 Jan;25(1):28-42.
16. Kangra K, Singh J. Comparative analysis of predictive machine learning algorithms for diabetes mellitus. Bulletin of Electrical Engineering and Informatics. 2023 Jun 1;12(3):1728-37.
17. Chicco D, Warrens MJ, Jurman G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. Peerj computer science. 2021;7:e623.
18. Chicco D, Warrens MJ, Jurman G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. Peerj computer science. 2021 Jul 5;7:e623.
19. Ahmed MS, Fakhrudeen AM. COVID-19IraqKirkukDataset: Development and evaluation of an Iraqi dataset for COVID-19 classification based on deep learning. International Journal of Nonlinear Analysis and Applications. 2023 Jan 1;14(1):2507-18.
20. Murti RP, Putra SM, Kurniawan SA, Nugraha YR. Naïve Bayes classifier for journal quartile classification. 2019.