Optimizing diabetes prediction using machine learning: a random forest approach

Aone Maenge, Tshiamo Sigwele, Cliford Bhende, Chandapiwa Mokgethi, Venumadhav Kuthadi, Blessing Omogbehin
Department of Computing and Informatics, Faculty of Pure and Applied Sciences, Botswana International University of Science and Technology, Palapye, Botswana

Article Info

Article history:

Received Dec 15, 2024
Revised Apr 11, 2025
Accepted Apr 23, 2025

Keywords:

Diabetes
Hospital Frankfurt Germany
Machine learning
Random diabetes prediction
Random forest

ABSTRACT

Diabetes, a leading cause of global mortality, is responsible for millions of deaths annually due to complications such as heart disease, kidney failure, and stroke. Projections indicate that 700 million people will be affected by diabetes in 2045, placing immense strain on global healthcare systems. Early detection and accurate prediction of diabetes are essential in mitigating complications and reducing mortality rates. However, existing diabetes prediction frameworks face challenges, including imbalanced datasets, overfitting, inadequate feature selection, insufficient hyperparameter tuning, and lack of comprehensive evaluation metrics. To address these challenges, the proposed random forest diabetes prediction (Random DIP) framework integrates advanced techniques such as hyperparameter tuning, balanced training, and optimized feature selection using a random search cross-validation (RandomizedSearchCV). This framework significantly improves predictive accuracy and ensures reliable clinical applicability. Random DIP achieves 99.4% accuracy, outperforming related works by 7.23%, the area under curve (AUC) of 99.6%, surpassing comparable frameworks by 7.32%, a recall of 100%, exceeding existing models by 9.65%, a precision (97.8%), F1-score (98.9%), and outperformance of 6.69%. These metrics demonstrate Random DIP's excellent capacity to identify diabetes cases while minimizing false negatives (FPs) and providing reliable predictions for clinical use. Future work will focus on integrating real-time clinical data and expanding the framework to accommodate multi-disease prediction for broader healthcare applications.

Corresponding Author:

Tshiamo Sigwele
Department of Computing and Informatics, Botswana International University of Science and Technology
Plot 10071, Boseja, Palapye, Botswana
Email: sigwelet@biust.ac.bw

1. INTRODUCTION

Diabetes mellitus is a chronic metabolic disorder that keeps the blood sugar level high because the body either does not produce enough insulin or does not use it correctly and can cause serious harm to many other organs, such as the heart, eyes, nerves, and even death [1], [2]. Diabetes has two main subtypes, namely type 1 diabetes (T1D) and type 2 (T2D), each requiring personalized interventions [3]. The T1D affects 10% of the world’s population while the remaining 90% is affected by T2D [4], [5]. It is very crucial to accurately diagnose these subtypes on time to avoid complications or death. Studies indicate that T2D patients with an early and accurate diagnosis may avoid 80% of complications [6]. Diabetes has affected over 422 million people globally, resulting in about 1.5 million deaths yearly [7]. According to estimates, 700 million people will be affected by the disease in 2045 worldwide [8]. According to WHO, Africa has over 24 million adults living with diabetes, and this number is estimated to increase by 129% to reach 55 million by 2045. These high mortality numbers indicate the urgent need for effective diabetes prediction frameworks for early diagnosis and prevention. Several machine learning (ML) frameworks have been proposed for diabetes predictions to obtain hidden insights from biomedical datasets to minimize diabetes complications at an early stage. Nevertheless, there exist critical gaps in current works that need to be addressed.

Research gaps: current ML frameworks rely on a minimal set of features, in this case, just five, which may make it more difficult for the model to accurately represent the complexity of diabetes-related factors. The exclusive reliance on lifestyle-related factors neglecting other crucial contributors to diabetes can potentially compromise the framework's comprehensiveness. The use of female-only datasets in model training introduces gender bias, potentially compromising the model's predictive accuracy and generalizability to underrepresented groups, such as males. In addition, the majority of frameworks are based solely on accuracy metrics overlooking other essential aspects of model performance. Current models exhibit suboptimal performance, characterized by low accuracy and high error rates, with some lacking documented accuracy metrics. A significant research gap exists in the lack of embedded-based feature selection methods for identifying critical data features, as well as the need for fine-tuning classifiers to enhance model accuracy. These observations emphasize the importance of addressing these limitations in developing and evaluating diabetes prediction frameworks to enhance their comprehensiveness, robustness, and applicability. Thus, it is essential to develop a framework that can predict diabetes in a feasible, precise, and cost-efficient manner. This research proposes the development of a ML framework for predicting diabetes accurately leveraging random forest algorithms to bridge gaps in existing diabetes frameworks. The contributions of this research work are as follows,

i) Gap analysis: identified key gaps in ML-based diabetes prediction frameworks include imbalanced and biased datasets, insufficient training data, overfitting, redundant and irrelevant features, inadequate feature selection, inadequate model tuning, neglect of comprehensive evaluation metrics, and suboptimal performance like predictive accuracy.

ii) Framework development: developed a random forest-based ML framework to predict diabetes called random forest diabetes prediction (Random DIP) to enhance prediction accuracy.

iii) Dataset manipulation: adopted and manipulated the Hospital Frankfurt dataset which included eight independent variables and one target variable to suit the Random DIP model.

iv) Evaluation: the proposed Random DIP framework significantly outperformed related works when evaluated for performance in terms of accuracy, area under curve (AUC), precision, recall, and F1-score.

The rest of this article is structured as follows: section 1 provides the Introduction of the research followed by section 2 which describes the proposed Random DIP framework. The study findings are presented and analyzed in section 3 while section 4 brings the study to a conclusion.

Literature review: we provide an in-depth gap analysis by conducting a review of the existing literature from 2024 up to 5 years ago on diabetes prediction, highlighting the limitations and research gaps. The gap analysis summary is that current ML frameworks for diabetes prediction face several gaps, including overfitting, feature redundancy, irrelevant features, imbalanced and biased datasets, insufficient data, neglect of performance metrics, suboptimal accuracy, and inadequate feature selection and tuning. The following are some of the detailed related frameworks with their contributions and gaps. Atif *et al.* [4] performs an analysis of ML classifiers for predicting diabetes mellitus in the preliminary stage but there is poor accuracy performance. Pranto *et al*. [5] analyzed diabetes prediction using the random forest algorithm but faced several limitations. The reliance on only four features draws attention to limited and inadequate feature selection, reducing the model’s ability to represent the complexity of diabetes-related factors, which increases error rates and hinders predictive accuracy. The model’s relatively low accuracy (78%), despite a recall of 89% and F1-score of 84%, emphasizes suboptimal performance and overfitting. Additionally, training exclusively on female data introduces gender bias, limiting the model’s generalizability to diverse populations, thereby producing biased predictions and oversimplified decision boundaries that fail to capture real-world complexities. Ahamed *et al*. [8] employed the light gradient boosting machine (LGBM) algorithm for diabetes prediction, achieving an accuracy of 95.20%. While the study explored transformer-based learning for dataset enhancement, it relied solely on accuracy for evaluation, overlooking other critical performance metrics like precision, recall, and AUC. Although the use of NumPy, Seaborn, and MATLAB for analysis provided transparency, the absence of further fine-tuning for classifiers reflects inadequate model optimization, limiting the opportunity to achieve even better performance. The study indicates the importance of utilizing diverse metrics and additional tuning to improve model evaluation and accuracy. Joshi and Dhakal [9] developed a diabetes prediction model using logistic regression (LR) and decision tree (DT) but encountered significant limitations. The use of only five features indicates potential redundancy and irrelevant features, restricting the model’s capacity to capture complex diabetes predictors. The exclusive reliance on data from women (Pima Indian dataset) introduces bias and imbalance, limiting generalizability. Reporting only accuracy and cross-validation error rate reflects neglect of comprehensive performance metrics, while the 78.26% accuracy and 21.74% error suggest suboptimal performance and potential overfitting. Unspecified tools and inadequate model details further hinder replicability and improvement opportunities. Aftab *et al*. [10] proposed a fused diabetes prediction model combining naïve Bayes, DT, and artificial neural network algorithms, achieving high accuracy (95.20%) with a miss rate or false negative (FN) rate) of 4.80%. However, the evaluation metrics were limited to accuracy and miss rate, neglecting comprehensive performance metrics such as recall, precision, and F1-score, which are essential for assessing broader model performance. Furthermore, the lack of details about the ML tools used reduces replicability and interpretability. These limitations, despite promising results, indicate the need for deeper evaluations and explicit tool specifications to ensure the robustness of the model. Saxena *et al*. in [11] predicted diabetes using the random forest algorithm with feature selection methods, achieving 79.83% accuracy, a specificity of 71.4%, a sensitivity of 79.8%, and an AUC of 83.6%. However, the model was trained exclusively on data from pregnant women in the Pima Indians dataset, introducing gender and population bias and limiting generalizability to broader demographics. While performance metrics such as sensitivity and AUC were promising, the relatively low accuracy indicates suboptimal performance. Additionally, the use of Weka 3.9 was documented, but the limited dataset diversity restricts the model's ability to make unbiased and representative predictions. Agliata *et al*. [12] developed a type 2 diabetes prediction model using the Adam algorithm, achieving an accuracy of 86% and a receiver operating characteristic (ROC) AUC of 93.4%. Chou *et al.* [13] proposes a framework predicting the onset of diabetes with ML methods. Taha and Malebary [14] proposes a hybrid meta-classifier of fuzzy clustering and logistic regression for diabetes prediction. Islam *et al.* [15] proposes a comparative approach to alleviating the prevalence of diabetes mellitus using ML. Anbananthen *et al.* [16] proposed a comparative performance analysis of hybrid and classical ML methods in predicting diabetes. Despite the strong ROC AUC, the evaluation relied solely on accuracy and AUC, neglecting comprehensive metrics such as sensitivity, specificity, and F1-score. The model utilized three datasets. While the dataset diversity adds value, the limited evaluation metrics restrict a holistic assessment of the model's effectiveness. This calls for broader metrics to provide more robust and interpretable model insights.

2. RESEARCH METHOD

This section describes the steps carried out in the development of the Random DIP model to address the identified gaps from the literature of overfitting, feature issues, biased datasets, insufficient data, limited performance metrics, suboptimal accuracy, inadequate feature selection, and tuning in ML models. Addressing these gaps will be evident through the improvement of performance metrics such as accuracy, AUC, precision, recall, and F1-score which correlate with the gaps. Figure 1 shows the architecture for the proposed Random DIP framework for diabetes prediction. The steps in Figure 1 are carried out to systematically build, train, and evaluate the proposed Random DIP framework using the publicly available Hospital Frankfurt Germany dataset. The proposed random forest framework is designed to achieve high prediction accuracy by leveraging ensemble learning techniques. This framework integrates advanced preprocessing, feature selection, hyperparameter tuning, and rigorous evaluation metrics to ensure robust and reliable predictions. In the following, we provide a detailed explanation of each phase of the methodology, accompanied by relevant equations where necessary.

2.1. Dataset acquisition

Dataset description and quality: the data acquisition phase is critical in developing the proposed diabetes prediction framework. This phase involves sourcing and validating a dataset containing features indicative of diabetes. The dataset used in this framework is the Hospital Frankfurt Germany dataset, which is publicly available on the Kaggle platform. The Hospital Frankfurt Germany dataset is chosen for its comprehensive feature set that captures critical diabetes indicators, making it highly relevant to the prediction task. Its large sample size enhances the model's ability to generalize across diverse patient populations, ensuring robust and reliable predictions. Additionally, its widespread adoption in previous research frameworks validates its credibility and utility in diabetes-related studies, reinforcing its suitability for the proposed framework [3], [10], [13], [14].

Dataset composition: the dataset contains 2,000 instances, with a distribution of 684 diabetic cases (34.2%) and 1,316 non-diabetic cases (65.8%). This balanced distribution ensures fair representation of both diabetic and non-diabetic classes, providing a solid foundation for training and testing predictive models while minimizing bias in classification results. This composition makes the dataset reliable for building accurate and balanced prediction algorithms.

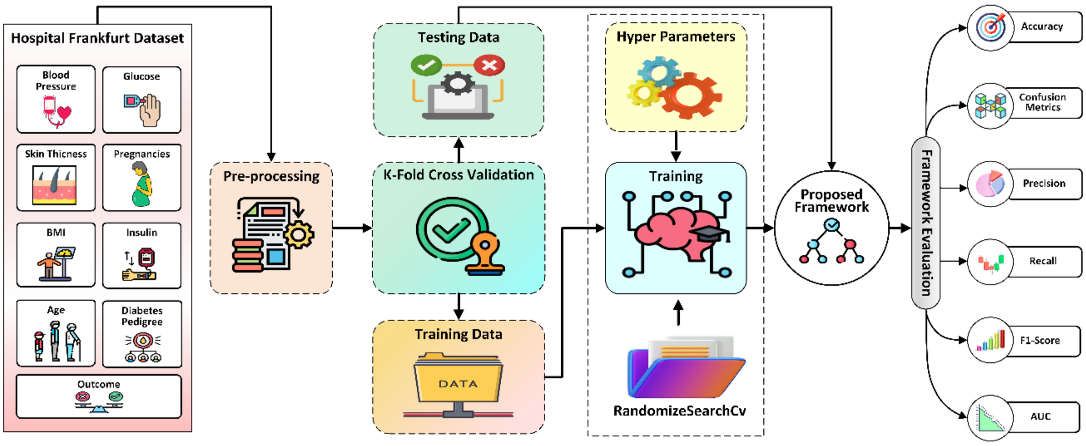

Figure 1. The proposed Random DIP framework architecture for the diabetes prediction

Dataset representation and descriptive characteristics: the dataset is shown as input to the framework in Figure 1. The data representation section explains how the dataset is structured, including feature organization, labels, and overall format, ensuring clarity for ML model training and evaluation. The dataset D is represented as (1).

$$D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \tag{1}$$

Where $x_i = \{x_{(i1)}, x_{(i2)}, \ldots, x_{im}\}$ denotes the feature vector for the i^{th} patient. The variable $y_i \in \{0,1\}$ is the dichotomous variable representing two possible where $y_i = 1$ if the patient has diabetes or $y_i = 0$ if the patient does not have diabetes. The variable $n = 2000$ is the total number of patient records and finally, $m = 9$ is the number of features in the dataset. These features include glucose levels, body mass index (BMI), insulin levels, age, blood pressure, skin thickness, pregnancies, diabetes pedigree function, and outcome, with the outcome variable indicating whether a patient is diabetic or not.

Table 1 presents descriptive statistics of the dataset. The average glucose level is 121.18 mg/dL, with a standard deviation of 32.07, indicating significant variability. The average BMI is 32.19, suggesting an overweight population and insulin levels have a mean of 80.25, with outliers such as a maximum value of 744. Patient ages range from 21 to 81 years, with a mean of 33.09 years. The dataset is balanced, with 34% diabetic cases, ensuring a reliable foundation for predictive analysis.

Table 1. The descriptive statistics of the Hospital Frankfurt Germany dataset

Statistic	Pregnancy	Glucose	Blood pressure	Skin thickness	Insulin	BMI	Pedigree function	Age	Outcome
Count	2,000	2,000	2,000	2,000	2,000	2,000	2,000	2,000	2,000
Mean	3.703	121.	69.1	20.9	80.25	32.19	0.471	33.10	0.342
Std	3.3063	32.06	19.18	16.10	111.2	8.141	0.323	11.79	0.474
Min	0.0	0.0	0.0	0.0	0.0	0.0	0.078	21.0	0.0
25%	1.0	99.0	63.5	0.0	0.0	27.375	0.244	24.0	0.0
50%	3.0	117.0	72.0	23.0	40.0	32.3	0.376	29.0	0.0
75%	6.0	141.0	80.0	32.0	130.0	36.8	0.624	40.0	1.0
Max	17.0	199.0	122.0	110.0	744.0	80.6	2.42	81.0	1.0

2.2. Data pre-processing

The data pre-processing phase is crucial for preparing the dataset for the random forest model. It involves a series of steps to ensure that the data is clean, relevant, and ready for analysis. These steps help enhance the quality of the dataset and, in turn, improve the model's performance. The phase includes exploratory data analysis (EDA), which helps uncover patterns and relationships within the data. It also covers techniques like handling missing values, normalizing features, detecting and removing outliers, and performing dimensionality reduction. The subsequent subsections explain these steps in detail, elaborating on each process and its importance in ensuring the dataset is optimal for training.

Data pre-processing phase 1: EDA is performed to summarize and visualize the dataset, providing insights into its structure and revealing patterns, correlations, or anomalies [15]. Feature distributions are examined using histograms and box plots to detect skewness, outliers, and missing values. Relationships between features are analyzed through scatter plots and correlation heatmaps. The correlation coefficient r quantifies the strength of relationships. Strong correlations (r >0.7) suggest redundancy, guiding feature selection for diabetes prediction. The formula for the correlation coefficient is (2).

$$r = \frac{Cov\ (X,Y)}{\sigma_X \sigma_Y} \tag{2}$$

Where $Cov\ (X, Y)$ is the covariance between variables X and Y, and σ_X, σ_Y are their respective standard deviations. In the Hospital Frankfurt Germany diabetes dataset as shown in Figure 2, higher glucose levels show a strong correlation with diabetes presence (r >0.5), while BMI and age have weaker associations (r between 0.2 to 0.3). Higher insulin levels correlate strongly with glucose, and higher skin thickness correlates with insulin levels. A higher BMI is weakly associated with diabetes and blood pressure, and older age shows a weak link to diabetes risk [16].

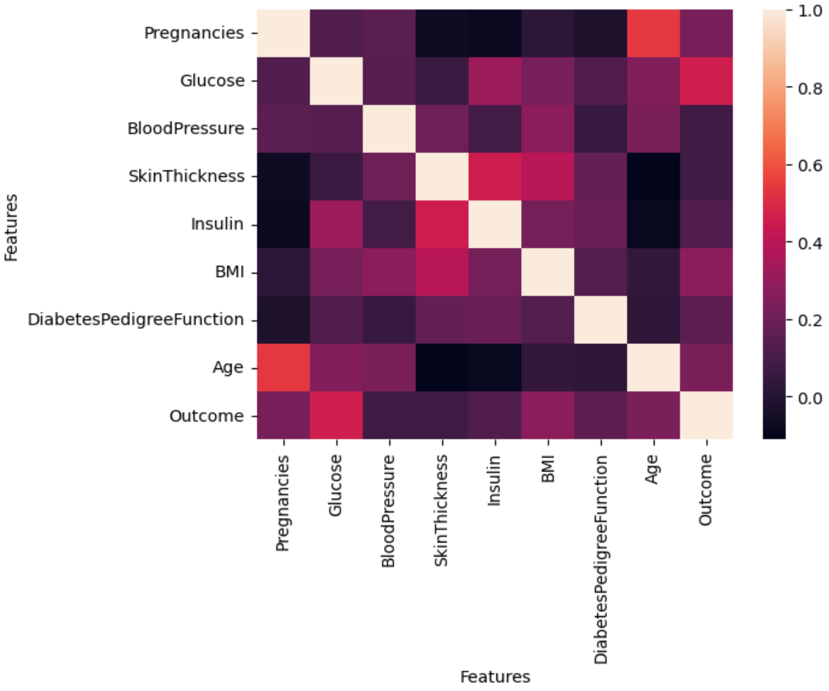

Figure 2. The correlation between features of Hospital Frankfurt Germany diabetes dataset

Data pre-processing phase 2: handling missing values is critical for improving model performance and ensuring accurate predictions. Missing values are imputed using the median value of the corresponding feature to avoid distortion from outliers [15]. The imputation formula is (3).

$$x_{ij} = \begin{cases} median\ \left(\{x_{1j}, x_{2j}, \dots, x_{nj}\}\right) & if\, x_{ij}\ is\ missing \\ x_{ij} & otherwise. \end{cases} \tag{3}$$

Here, x_{ij} is the value of the j-th feature for the i-th sample, and the median $\left(\{x_{1j}, \dots, x_{nj}\}\right)$ is the median of the feature across all samples. This method ensures the dataset remains robust without introducing biases [16].

Data pre-processing phase 3: feature normalization is applied to scale features to a comparable range, ensuring that large-magnitude features do not dominate model training. The z-score normalization formula is (4).

$$x'_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{4}$$

Where x'_{ij} is the normalized value, x_{ij} is the original value, μ_j is the mean of the j-th feature, and σ_j is its standard deviation. This standardization centers each feature around a mean of 0 with a unit standard deviation, enhancing model convergence and improving performance [16].

Data pre-processing phase 4: outlier detection and removal, the outliers shown in Figure 3 are data points that deviate significantly from the rest of the dataset, often caused by errors in data collection or measurement. These outliers can distort predictions and lead to inaccurate model performance. In diabetes prediction, abnormal values, such as extreme glucose levels, can skew results, making the model unreliable. To address this, the interquartile range (IQR) method is used to detect and remove outliers. The IQR and outlier inequality are calculated as (5) and (6).

$$IQR = Q_3 - Q_1 \tag{5}$$

$$x_{ij} < Q_1 - 1.5 \cdot IQR \ \ or \ \ x_{ij} > Q_3 + 1.5 \cdot IQR \tag{6}$$

Where Q_1 and Q_3 represent the 25th and 75th percentiles of the dataset, respectively. Any data point x_{ij} falling outside the range in (6) is considered an outlier and removed. This process ensures cleaner, more reliable data, improving model generalization and prediction accuracy. Removing outliers as shown in Figure 4 helps the model avoid instability, overfitting, and poor performance, leading to better decision-making [16]-[18].

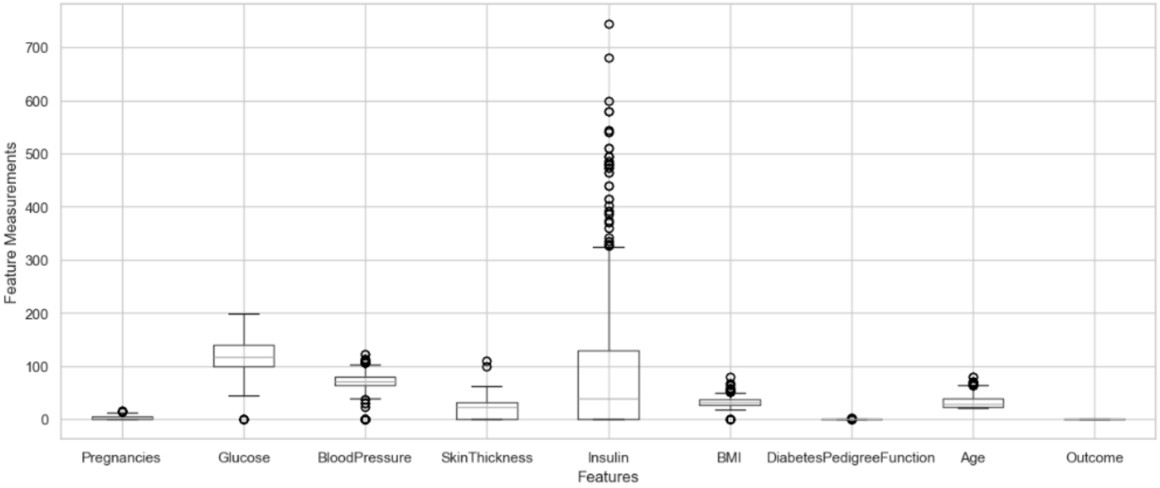

Figure 3. Data pre-processing phase 4: outlier detection of each feature

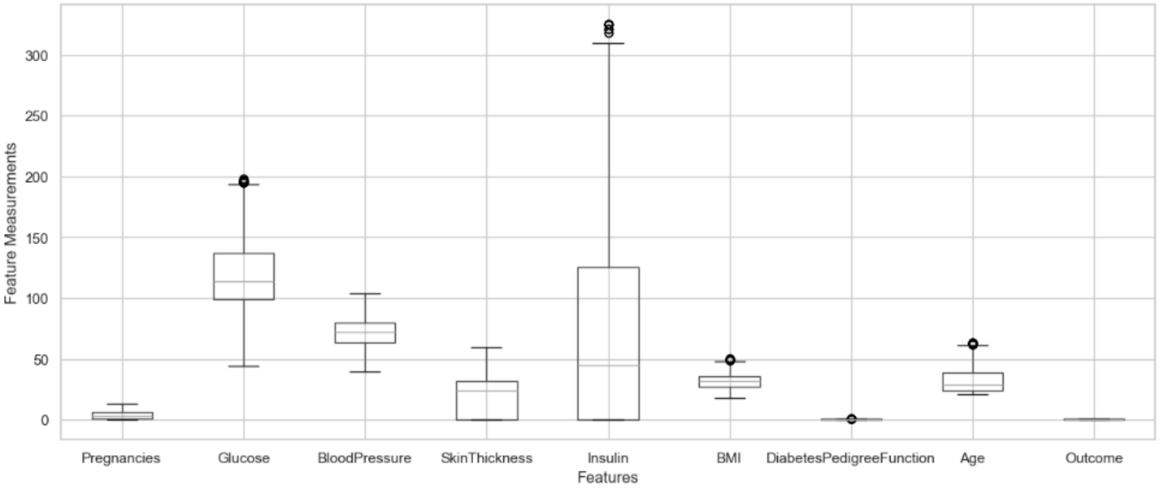

Figure 4. Data pre-processing phase 4: outliers removal data points distribution

Data pre-processing phase 5: dimensionality reduction, to reduce computational complexity and mitigate overfitting, principal component analysis (PCA) can be applied. PCA transforms the original data matrix X into a lower-dimensional space Z while retaining most of the data variance $Z = XW$, where X is the original data matrix, W is the matrix of eigenvectors (principal components) derived from the covariance matrix of X, and Z is the transformed feature space. For instance, high-dimensional features like BMI, insulin levels, and glucose measurements are condensed into fewer dimensions while preserving critical patterns influencing diabetes prediction [16]. This reduces computational complexity and mitigates the risk of overfitting.

2.3. Model training

Model training phase 1: data splitting, when we train the random forest model, we are teaching it to predict the outcome (whether a person has diabetes or not) based on patterns in the training data. This training process allows the model to learn from the features (such as glucose levels and BMI) and make predictions for new, unseen data. Once the data has been pre-processed and cleaned, it is split into training and testing sets using an 80:20 ratio. The training set (80%) is used to train the ML model, while the testing set (20%) is used to evaluate its performance. This ensures the model can generalize to new, unseen data. Tools like scikit-learn's train_test_split function are used to randomly divide the dataset, maintaining a balanced representation of diabetic and non-diabetic cases in both subsets, which helps improve model accuracy and reliability. The training set is used to train the random forest model, a ML algorithm designed to predict diabetes outcomes. During training, the model learns the patterns and relationships between the input features (such as glucose levels, insulin, and BMI) and the target variable (diabetes status). After training, the model is evaluated using the testing set, which contains data it has never seen before. Evaluation metrics such as accuracy, precision, recall, and F1-score are calculated to assess the model's predictive performance. Additionally, results from K-fold cross-validation are used to fine-tune hyperparameters such as tree depth and number of estimators to improve the model’s generalization and accuracy, ensuring it performs optimally on new, unseen data. The following areare the detailed phases for model training of Random DIP.

Model training phase 2: algorithm for creation of random forest, the random forest algorithm, as outlined in Algorithm 1, is employed to construct a robust diabetes prediction model. This ensemble learning approach creates multiple DTs, each trained on a random subset of the diabetes dataset using the bootstrap sampling method [3], [19]. The model predicts diabetes outcomes by aggregating predictions from all the individual DTs. The prediction process for the random forest model is represented mathematically as (7).

$$\gamma = mode\ (T_1(x), T_2(x), \dots, T_T(x)) \tag{7}$$

Where γ is the predicted diabetes classification result for the input features $x, T_i(x)$ denotes the prediction from the i-th DT, and T is the total number of DTs in the ensemble. The mode function aggregates predictions by selecting the most frequently occurring class label across all trees. This majority voting mechanism ensures that the model reduces overfitting compared to individual DTs [20], [21]. By combining the strengths of multiple trees, random forest enhances predictive accuracy and generalization, making it an effective tool for diabetes classification. The mode function aggregates predictions by selecting the most frequently occurring class label across all trees.

Algorithm 1. Algorithm for creation of random forest
Input: no. of trees (T), no. of features (m), training dataset (X_{train}, y_{train}), bootstrap sampling method.
Output: γ: final prediction (diabetes classification result).

```
Set no. of trees: define the total number of DTs for diabetes prediction as T.
Select no. of features: specify $F_m$, the number of input features used by each tree to split nodes.
Initialize counter: set tree counter i←1.
while i ≤T do
    Randomly sample data with replacement from the diabetes training dataset $D_{train}$.
    Randomly select $F_m$, the subset of features for the $i^{th}$ tree from the total feature set.
    Train the $i^{th}$ DT $T_i$ using the sampled dataset and selected feature subset.
    Increment tree counter: i←i+1.
end
Final prediction: determine diabetes outcome using majority voting across T trees for new inputs
```

Model training phase 3: K-fold cross-validation enhances the robustness and generalizability of the diabetes prediction model by dividing the dataset into K equal-sized subsets or folds. The model is trained K times, using K-1 folds for training and the remaining fold for testing. This method ensures a more reliable evaluation of the model's performance and mitigates overfitting. The performance metric for each fold (C_i) is computed, and the average performance (CV_{avg}) is calculated across all K folds. This approach ensures that each subset of the diabetes dataset is used for testing, offering a comprehensive assessment of the model's ability to predict diabetes accurately.

$$CV_{avg} = \frac{1}{K}\sum_{i=1}^{K} CV_i \tag{8}$$

Where CV_{avg} represents the average performance of the model across all folds, providing an overall evaluation of the model's ability to predict diabetes. K denotes the total number of folds or subsets of the divided dataset. CV_i refers to the performance metric (e.g., accuracy, precision, and recall) obtained from the i^{th} fold during testing, which reflects how well the model performs on that specific subset. By averaging CV_i values across all folds, the model's generalizability is assessed, ensuring it performs well on unseen data.

Model training phase 4: hyperparameter optimization, in this section, the training process of the random forest model is integrated with hyperparameter tuning to optimize its performance for diabetes prediction, as shown in Algorithm 2. Random search cross-validation (RandomizedSearchCV) is chosen as an effective technique for finding the best combination of hyperparameters for the random forest model, optimizing it for better predictive performance. It helps in tuning key parameters such as the number of trees (T), maximum depth (max_depth), the number of features used in tree splitting (m), minimum samples required to split an internal node (min_samples_split), and the minimum samples required to be at a leaf node (min_samples_leaf). This tuning directly influences the model's accuracy and ability to generalize. The model training consists of fitting the random forest algorithm using the training data using Algorithm 1, while simultaneously fine-tuning the hyperparameters using RandomizedSearchCV. The objective is to maximize the performance metric M (such as accuracy, precision, or recall) by adjusting these parameters, which enhances the random forest model's ability to predict diabetes. The optimization problem is expressed as (9).

$$\Theta^* = arg \max_{\Theta \in H} M(F_{RF}(D_{train}, \Theta), Y_{train}) \tag{9}$$

Where Θ represents the set of hyperparameters, which includes the number of trees (T), maximum depth (max_depth), and the number of features (m), used for splitting. The variable H is the hyperparameter grid that defines the possible combinations of these parameters. The variable $F_{RF}(D_{train}, \Theta)$ is the random forest model trained on diabetes training data D_{train} with the hyperparameters Θ. The variable M is the performance metric (e.g., accuracy, precision, and recall) used to evaluate the model's ability to predict diabetes. The variable Y_{train} is the actual label of diabetes in the training set. The variable Θ* is the optimal set of hyperparameters that maximizes the performance metric.

Algorithm 2. Steps for RandomizedSearchCV to optimize hyperparameters

```
Input:  Hyperparameter grid (H), no. of iterations ($n_{iter}$), cross-validation folds (K), diabetes training data
        ($D_{train}$), diabetes test data ($D_{test}$), number of trees/estimators (T), number of features (m)
Output: Optimized hyperparameters ($\Theta^*$), trained random forest model ($F_{RF}$), diabetes predictions ($\hat{y}_{test}$),
        performance metrics (e.g., accuracy, precision, and recall).
1.  Define hyperparameter grid: define H, including T, max_depth, and m.
2.  Initialize RandomizedSearchCV: set up H, iterations $n_{iter}$, and K-fold CV.
3.  Train models on diabetes data: fit RandomizedSearchCV using diabetes training data $D_{train}$.
4.  Select best hyperparameters: choose optimal $\Theta^*$ maximizing CV accuracy for classification.
5.  Final diabetes prediction: optimized $\Theta^*$ to predict diabetes outcomes on test data.
```

3. RESULTS AND DISCUSSION

3.1. Model evaluation metrics

Table 2 shows a summary of the adopted evaluation metrics, their equations, and their definition in diabetes prediction terms. Evaluation is a crucial stage in the ML process. Predictions are made on a 20% test dataset using the previously trained framework. This step assesses the framework's ability to generalize new data and measures its effectiveness in practical situations. The primary objective is to evaluate the accuracy and robustness of the trained framework when applied to unseen data. Evaluation helps identify potential issues like overfitting or underfitting and provides insights into the framework's generalization capabilities. To effectively assess the impact of the algorithm, it is essential to define specific performance metrics that can measure the quality of a classification framework [4]. The model evaluation metrics are accuracy, precision, recall, F1-score, and ROC. The performance evaluation primarily involves calculations based on the confusion matrix [2]. A confusion matrix evaluates how well a classification framework predicts diabetic and non-diabetic patients as follows [2], [4]. A true positive (TP) shows that a diabetic patient is correctly predicted as diabetic. A true negative (TN) shows that a non-diabetic patient is correctly predicted as non-diabetic. A FN shows that a diabetic patient is incorrectly predicted as non-diabetic. Lastly, a false positive (FP) indicates that a non-diabetic patient is incorrectly predicted as diabetic. The framework development and evaluation are conducted using Python version 3.12 programming language on Jupyter Notebook version 6.5.4 on a 64-bit Windows 10 operating system. The hardware specifications include an Intel(R) Core (TM) i3-7020U CPU @2.30GHz and 4.00 GB of internal RAM.

Table 2. The adopted evaluation metrics

Metric	Definition	Equation	
Accuracy	Calculates the proportion of correctly classified patients (both diabetic and non-diabetic) out of all patients, giving an overall performance indicator of the framework [13].	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$	(1)
Precision	Evaluates the proportion of correctly identified diabetic patients (TPs) out of all patients predicted to be diabetic [22].	$Precision = \frac{TP}{TP + FP}$	(2)
Recall	Measures the proportion of correctly identified diabetic patients (TPs) among all actual diabetic patients. It indicates the framework's ability to capture all diabetic cases [17].	$Recall = \frac{TP}{TP + FN}$	(3)
F1-score	Combines the framework's ability to avoid falsely labeling non-diabetic patients as diabetic (precision) and its effectiveness in correctly identifying actual diabetic patients (recall) [4], [15].	$F1Score = \frac{2x(Precision\ x\ Recall)}{Precision + Recall}$	(4)
ROC Curve (AUC)	Distinguish between diabetic and non-diabetic patients across different threshold values [23].	$AUC = \int_{1}^{0} TPR\left(FPR^{-1}(t)\right)dt$	(5)

3.2. Parameter grid

In the experiment, we defined a parameter grid, H, with several important hyperparameters for the random forest framework. The parameter n_estimators, which controls the number of trees in the forest, was tested with values ranging from 1 to 1,000. The max_features parameter, which determines the number of features to consider when making a split, was tested with options such as 'auto', 'sqrt', and 'log2'. The max_depth parameter, which sets the maximum depth of each tree, varied between 1 and 250 to strike a balance between capturing complex patterns and preventing overfitting. Additionally, we adjusted the min_samples_split and min_samples_leaf parameters, with values ranging from 2 to 5 and 1 to 3, respectively. These parameters control the minimum number of samples required to split a node and to be present at a leaf node, thereby influencing the framework's complexity and generalization ability.

3.3. Best framework parameters

The hyperparameter search identified an effective combination that significantly boosted model performance. A max_depth of 188 allowed trees to capture complex patterns, while max_features set to 'auto' enabled the use of all available features during splits. The min_samples_leaf was set to 1, allowing highly detailed trees, and the min_samples_split set to 3 helped prevent overfitting by requiring at least three samples to split a node. Additionally, n_estimators was set to 22, providing a compact yet strong ensemble. These optimized settings resulted in a best cross-validation score of 0.9719, indicating strong generalization to unseen data.

3.4. Proposed framework results-no comparison

The RandomizedSearchCV was set up to assess ten different combinations of parameters through 10-fold cross-validation [24], [25], resulting in a total of 100 framework fits. After this extensive search, the framework was tested on a separate test set. Figure 5 shows the results of the proposed framework. The results of the proposed Random DIP framework demonstrate its exceptional capability in predicting diabetes, with notable trends and patterns that highlight its effectiveness. The accuracy of 99.4% indicates that Random DIP is highly reliable in correctly identifying both diabetic and non-diabetic individuals. The high accuracy suggests that the model has learned to capture the underlying patterns in the data, ensuring minimal misclassification, which is critical in medical diagnosis to avoid FNs or FPs. The ROC AUC score of 99.6% suggests that Random DIP is highly proficient in distinguishing between diabetic and non-diabetic patients.

The near-perfect value reflects the model's ability to maintain high performance even when adjusting the decision threshold, ensuring that the prediction system does not miss patients with diabetes, a crucial aspect in early diagnosis and treatment.

A precision of 97.8% means that when Random DIP predicts a patient has diabetes, it is highly likely to be correct. This is crucial in healthcare because high precision reduces the occurrence of FPs, preventing patients from undergoing unnecessary medical treatments or interventions. The perfect recall score (100%) indicates that the model identifies all actual diabetic patients without missing any. This is especially important in diabetes prediction, as missing a diabetic patient could lead to delayed diagnosis and treatment, potentially resulting in severe health complications. The model's ability to achieve perfect recall indicates its effectiveness in catching every possible diabetes case, ensuring early intervention. F1-score (98.9%). The F1-score with a value of 98.9% reflects a well-balanced model. This high F1-score demonstrates that Random DIP not only performs well in identifying diabetic cases but also maintains a strong ability to avoid FPs, making it an ideal model for practical diabetes prediction.

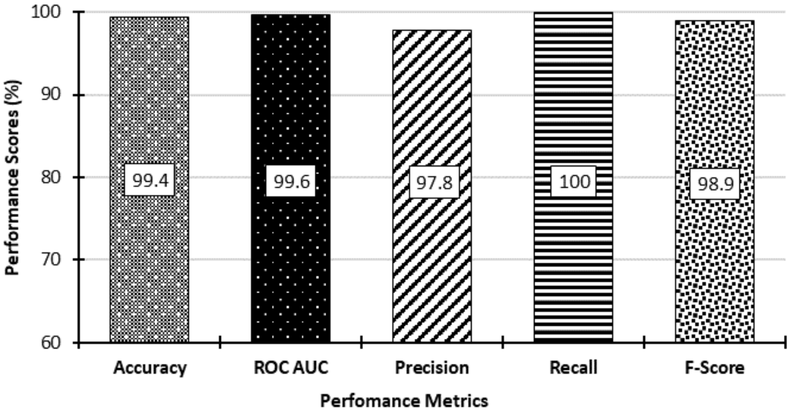

Figure 5. Proposed framework results

3.5. Proposed framework results–comparison with related frameworks

The proposed framework is now compared with related works from the reviewed. The comparison frameworks are Atif *et al.* [4], Chou *et al.* [13], Anbananthen *et al.* [16] shortened as Anban, and our proposed Random DIP. We compared our framework with existing frameworks for performance in terms of the evaluation metrics of accuracy, ROC AUC, precision, recall, and F1-score. We did this for a fair comparison, as our framework uses the same metrics. Our comparison results clearly show that the proposed random forest framework outperforms other methods in all evaluated metrics.

Figure 6 shows the accuracy metric of all the frameworks. The figure highlights that the proposed Random DIP framework significantly outperforms all other frameworks with an accuracy of 99.4%. The proposed Random DIP outperforms Chou *et al.* [13] (95.3%) by 4.30%, Atif *et al.* [4] (97.2%) by 2.26%, and Anbananthen *et al.* [16] (98.5%) by 0.91%. The reason for the high accuracy of the proposed framework compared to others is due to (1) effective hyperparameter tuning through RandomizedSearchCV, which optimizes the random forest model's parameters, and (2) robust feature selection that eliminates irrelevant variables and enhances model performance. The high accuracy means that the proposed framework is highly reliable in classifying patients correctly as diabetic or non-diabetic. This high accuracy metric solved the research gaps of imbalanced and biased datasets, insufficient training data, and suboptimal predictive accuracy in existing frameworks.

Figure 7 illustrates the ROC AUC metric for all frameworks. The Random DIP framework achieves an impressive ROC AUC of 99.6%, demonstrating its superior ability to distinguish between diabetic and non-diabetic cases. The proposed Random DIP outperforms Atif *et al.* [4] (97.2%) by 2.47%, Anbananthen *et al.* [16] (98.3%) by 1.32%, and Chou *et al.* [13] (99.1%) by 0.50%. The reason for the high ROC AUC of the proposed framework is (1) comprehensive data preprocessing, which ensures clean and unbiased input data, and (2) optimized DTs within the random forest model, leading to better separation of diabetic and non-diabetic cases. The high ROC AUC means that the framework can reliably differentiate TPs and TNs across varying decision thresholds. This metric of high ROC AUC solved the research gaps of overfitting, inadequate feature selection, and neglect of comprehensive evaluation metrics.

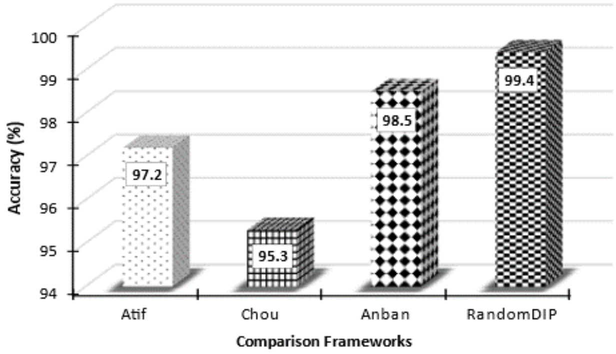

Figure 6. Comparison in terms of accuracy metric

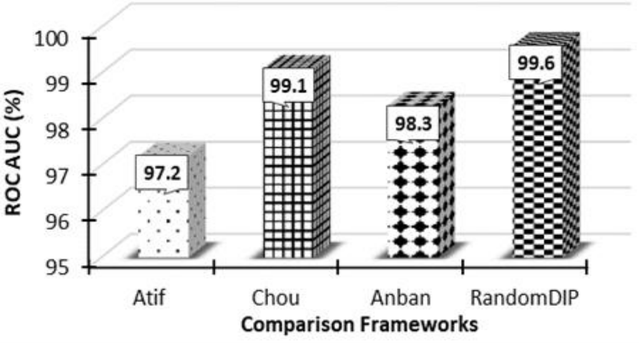

Figure 7. Comparison in terms of AUC metric

Figure 8 displays the precision metric across the frameworks. The proposed Random DIP framework achieves a precision of 97.8%, reflecting its accuracy in correctly identifying positive diabetes cases among all predicted positives. The proposed Random DIP outperforms Chou *et al.* [13] (92.7%) by 5.50%, Atif *et al.* [4] (97.2%) by 0.62%, and performs slightly under Anbananthen *et al.* [16] (98.8%) by −1.01%. The reason for the high precision of the proposed framework compared to others is (i) balanced data handling, which avoids bias during training and (ii) effective feature extraction, which improves the model's focus on relevant variables. The high precision means the proposed framework minimizes FPs, ensuring non-diabetic individuals are not misclassified as diabetic. The slightly lower precision of Random DIP (97.8% vs. 98.8%) is due to differences in handling FPs. Anbananthen *et al.* [16] stricter thresholds may improve precision, but Random DIP's balanced metrics and perfect recall ensure superior diabetes prediction overall. This metric of high precision solved the research gaps of redundant and irrelevant features, insufficient training data, and suboptimal performance like predictive accuracy.

Figure 9 represents the recall metric for the frameworks. The Random DIP framework achieves a perfect recall of 100%, outperforming Chou *et al.* [13] (93.1%) by 7.41%, Atif *et al.* [4] (97.2%) by 2.87%, and Anbananthen *et al.* [16] (98.8%) by 1.21%. The reason for the high recall of the proposed framework is (1) robust cross-validation techniques, which enhance the generalizability of the model, and (2) thorough hyperparameter optimization, which ensures that the decision boundaries capture all positive cases. The high recall means the framework is highly effective at identifying all true diabetic cases, reducing the risk of missed diagnoses. This metric of high recall solved the research gaps of imbalanced and biased datasets, neglect of comprehensive evaluation metrics, and inadequate model tuning.

Figure 10 shows the F1-score metric comparison. The Random DIP framework achieves an F1-score of 98.9%, showing its balanced performance in precision and recall. The proposed random DIP outperforms Atif *et al.* [4] (88.9%) by 11.24%, Chou *et al.* [13] (92.9%) by 6.46%, and Anbananthen *et al.* [16] (98.8%) by 0.10%. The reason for the high F1-score of the proposed framework is the advanced hyperparameter tuning, which ensures an optimal trade-off between precision and recall, and the effective feature engineering, which ensures the model is trained on the most relevant variables. The high F1-score means the framework balances precision and recall effectively, making it highly reliable in diabetes prediction tasks. This metric of high F1-score solved the research gaps of overfitting, inadequate feature selection, and suboptimal performance like predictive accuracy.

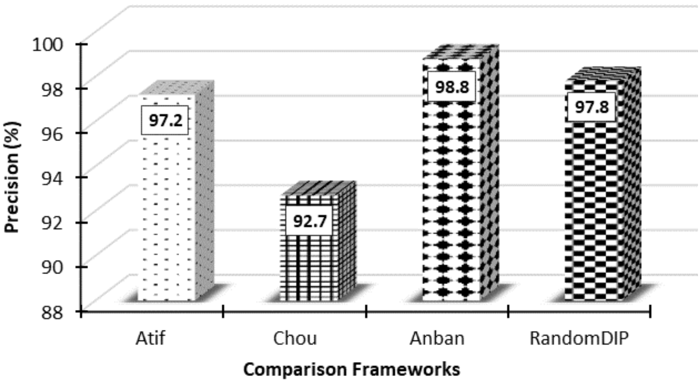

Figure 8. Comparison in terms of precision metric

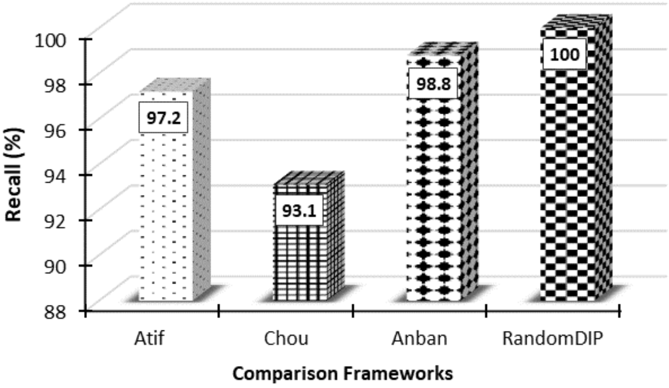

Figure 9. Comparison in terms of recall metric

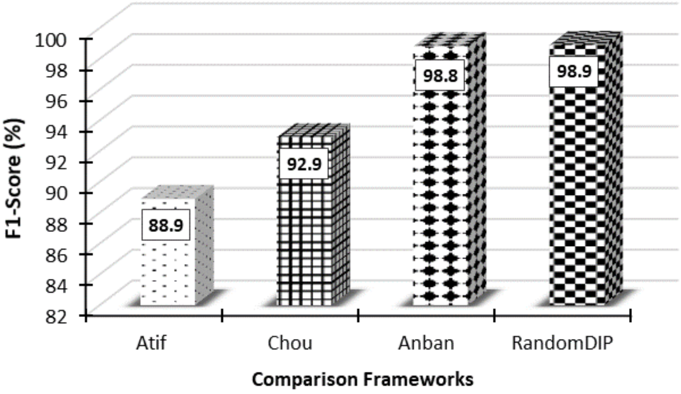

Figure 10. Comparison in terms of F1-score metric

4. CONCLUSION

Diabetes, a leading cause of global mortality, claims millions of lives annually through complications like heart disease, kidney failure, and stroke. Existing diabetes prediction frameworks face gaps such as imbalanced datasets, overfitting, inadequate feature selection, and insufficient hyperparameter tuning, limiting their reliability in clinical settings. The proposed Random DIP framework addresses these gaps through advanced hyperparameter tuning, balanced training techniques, and comprehensive evaluation across diverse metrics. By optimizing feature selection and employing RandomizedSearchCV, Random DIP reduces redundancy and enhances predictive accuracy. Random DIP achieves exceptional results with 99.4% accuracy, outperforming current works by 7.23%, 99.6% ROC AUC, surpassing related frameworks by 7.32%, 100% recall, exceeding existing frameworks by 9.65%, 97.8% precision, and 98.9% F1-score, outperforming comparable works by 6.69%. These results signify the framework's ability to identify diabetes cases accurately while minimizing FNs, crucial for clinical reliability. By addressing gaps in dataset balance, feature selection, and evaluation, Random DIP ensures robust diabetes prediction. Future improvements include optimizing precision to reduce FPs, integrating real-time clinical data for dynamic adaptability, and extending the framework for multi-disease prediction to broaden its healthcare impact.

ACKNOWLEDGEMENTS

The authors thank Botswana International University of Science and Technology for providing access to resources and facilities and extending heartfelt gratitude to their families for their unwavering support and encouragement.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Aone Maenge	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Tshiamo Sigwele	✓	✓		✓	✓			✓		✓	✓	✓	✓	
Clifford Bhende	✓		✓	✓			✓			✓	✓			
Chandapiwa Mokgethi	✓						✓			✓	✓			
Venu Madhav Kuthadi	✓					✓				✓	✓	✓	✓	
Blessing Omogbehin		✓								✓			✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis

I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting

Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] U. Ahmed *et al.*, "Prediction of diabetes empowered with fused machine learning," *IEEE Access*, vol. 10, pp. 8529–8538, 2022, doi: 10.1109/ACCESS.2022.3142097.

[2] P. Durga and T. Sudhakar, "An analysis of various machine learning techniques for predicting diabetes in its early stages," *Journal of Pharmaceutical Negative Results*, vol. 13, no. S01, Jan. 2022, doi: 10.47750/pnr.2022.13.S01.238.

[3] M. A. Rahim, M. A. Hossain, M. N. Hossain, J. Shin, and K. S. Yun, "Stacked ensemble-based type-2 diabetes prediction using machine learning techniques," *Annals of Emerging Technologies in Computing*, vol. 7, no. 1, pp. 30–39, 2023, doi: 10.33166/AETiC.2023.01.003.

[4] M. Atif, F. Anwer, F. Talib, R. Alam, and F. Masood, "Analysis of machine learning classifiers for predicting diabetes mellitus in the preliminary stage," *IAES International Journal of Artificial Intelligence*, vol. 12, no. 3, pp. 1302–1311, 2023, doi: 10.11591/ijai.v12.i3.pp1302-1311.

[5] B. Pranto, S. M. Mehnaz, E. B. Mahid, I. M. Sadman, A. Rahman, and S. Momen, "Evaluating machine learning methods for predicting diabetes among female patients in Bangladesh," *Information*, vol. 11, no. 8, 2020, doi: 10.3390/INFO11080374.

[6] K. Kangra and J. Singh, "Comparative analysis of predictive machine learning algorithms for diabetes mellitus," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 3, pp. 1728–1737, 2023, doi: 10.11591/eei.v12i3.4412.

[7] A. U. Haq *et al.*, "Intelligent machine learning approach for effective recognition of diabetes in e-healthcare using clinical data," *Sensors*, vol. 20, no. 9, 2020, doi: 10.3390/s20092649.

[8] B. S. Ahamed, M. S. Arya, and A. O. V. Nancy, “Diabetes mellitus disease prediction using machine learning classifiers with oversampling and feature augmentation,” *Advances in Human-Computer Interaction*, vol. 2022, 2022, doi: 10.1155/2022/9220560.

[9] R. D. Joshi and C. K. Dhakal, “Predicting type 2 diabetes using logistic regression and machine learning approaches,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 14, 2021, doi: 10.3390/ijerph18147346.

[10] S. Aftab, S. Alanazi, M. Ahmad, M. A. Khan, A. Fatima, and N. S. Elmitwally, “Cloud-based diabetes decision support system using machine learning fusion,” *Computers, Materials and Continua*, vol. 68, no. 1, pp. 1341–1357, 2021, doi: 10.32604/cmc.2021.016814.

[11] R. Saxena, S. K. Sharma, M. Gupta, and G. C. Sampada, “A novel approach for feature selection and classification of diabetes mellitus: machine learning methods,” *Computational Intelligence and Neuroscience*, vol. 2022, 2022, doi: 10.1155/2022/3820360.

[12] A. Agliata, D. Giordano, F. Bardozzo, S. Bottiglieri, A. Facchiano, and R. Tagliaferri, “Machine learning as a support for the diagnosis of type 2 diabetes,” *International Journal of Molecular Sciences*, vol. 24, no. 7, 2023, doi: 10.3390/ijms24076775.

[13] C. Y. Chou, D. Y. Hsu, and C. H. Chou, “Predicting the onset of diabetes with machine learning methods,” *Journal of Personalized Medicine*, vol. 13, no. 3, 2023, doi: 10.3390/jpm13030406.

[14] A. A. Taha and S. J. Malebary, “A hybrid meta-classifier of fuzzy clustering and logistic regression for diabetes prediction,” *Computers, Materials and Continua*, vol. 71, no. 2, pp. 6089–6105, 2022, doi: 10.32604/cmc.2022.023848.

[15] M. R. Islam, S. Banik, K. N. Rahman, and M. M. Rahman, “A comparative approach to alleviating the prevalence of diabetes mellitus using machine learning,” *Computer Methods and Programs in Biomedicine Update*, vol. 4, 2023, doi: 10.1016/j.cmpbup.2023.100113.

[16] K. S. M. Anbananthen, M. B. M. A. Busst, R. Kannan, and S. Kannan, “A comparative performance analysis of hybrid and classical machine learning method in predicting diabetes,” *Emerging Science Journal*, vol. 7, no. 1, pp. 102–115, 2023, doi: 10.28991/ESJ-2023-07-01-08.

[17] T. Daghistani and R. Alshammari, “Comparison of statistical logistic regression and randomforest machine learning techniques in predicting diabetes,” *Journal of Advances in Information Technology*, vol. 11, no. 2, pp. 78–83, 2020, doi: 10.12720/jait.11.2.78-83.

[18] M. K. Hasan, M. A. Alam, D. Das, E. Hossain, and M. Hasan, “Diabetes prediction using ensembling of different machine learning classifiers,” *IEEE Access*, vol. 8, pp. 76516–76531, 2020, doi: 10.1109/ACCESS.2020.2989857.

[19] H. N. Lakshmi, A. V. Vathsala, B. K. Upadhyay, and A. N. Rao, “Application and analysis of machine learning algorithms on pima and early diabetes datasets for diabetes prediction,” *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, pp. 28–35, 2023, doi: 10.17762/ijritcc.v11i5s.6594.

[20] S. Upadhyay and Y. K. Gupta, “Prediction of diabetes in adults using supervised machine learning model,” *Indian Journal of Engineering*, vol. 20, no. 53, 2023, doi: 10.54905/disssi/v20i53/e26ije1657.

[21] E. K. Oikonomou and R. Khera, “Machine learning in precision diabetes care and cardiovascular risk prediction,” *Cardiovascular Diabetology*, vol. 22, no. 1, 2023, doi: 10.1186/s12933-023-01985-3.

[22] Y. Qin *et al.*, “Machine learning models for data-driven prediction of diabetes by lifestyle type,” *International Journal of Environmental Research and Public Health*, vol. 19, no. 22, 2022, doi: 10.3390/ijerph192215027.

[23] S. K. Sharma *et al.*, “A diabetes monitoring system and health-medical service composition model in cloud environment,” *IEEE Access*, vol. 11, pp. 32804–32819, 2023, doi: 10.1109/ACCESS.2023.3258549.

[24] J. Shin *et al.*, “Development of various diabetes prediction models using machine learning techniques,” *Diabetes and Metabolism Journal*, vol. 46, no. 4, pp. 650–657, 2022, doi: 10.4093/dmj.2021.0115.

[25] F. Mohsen, H. R. H. Al-Absi, N. A. Yousri, N. El Hajj, and Z. Shah, “A scoping review of artificial intelligence-based methods for diabetes risk prediction,” *npj Digital Medicine*, vol. 6, no. 1, 2023, doi: 10.1038/s41746-023-00933-5.

BIOGRAPHY OF AUTHORS

Aone Maenge is an MSc student in the Department of Computing and Informatics at Botswana International University of Science and Technology. His research focuses on applying various machine learning models for diabetes prediction. He has been involved in several collaborations with other machine learning researchers in the department to expand and complement his knowledge and skill in the areas of research. He can be contacted at email: ma23018971@studentmail.biust.ac.bw.

Tshiamo Sigwele is currently a lecturer in the Department of Computing and Informatics at Botswana International University of Science and Technology (BIUST) with research interests in cloud computing, machine learning, and wireless communication. Dr. Sigwele graduated in 2017 with a Ph.D. in Cloud Computing And Telecommunications from the University of Bradford, UK. He has over 20 internationally recognized publications. He worked as a researcher from 2017 to 2018 in a British Council-funded project, BLESS U: Bandar Lampung Enhanced Smart Health Services with Smart Ubiquity, with a grant total of €89,937, and published several high-quality publications. He is currently supervising PhD and MSc students in the areas of cloud computing and machine learning. He is involved in several research projects at BIUST. He can be contacted at email: sigwelet@biust.ac.bw.

Clifford Bhende is an MSc student focusing on applying machine learning models for Ischemic predictions in the Department of Computing and Informatics at Botswana International University of Science and Technology. His focus research is in IHD prediction using machine learning techniques. He specializes in ischemic heart disease prediction using machine learning techniques by developing accurate predictive models to identify at-risk individuals, aiming to improve early detection and intervention strategies for better cardiovascular health outcomes. He is also involved in collaborations with other researchers in areas of machine learning in the department. He can be contacted at email: bk22100097@studentmail.biust.ac.bw.

Chandapiwa Mokgethi is an MSc student focusing on applying machine learning models in optimizing energy in mobile edge computing in the Department of Computing and Informatics at Botswana International University of Science and Technology. Her research focuses on cloud computing, edge computing, machine learning, and energy efficiency. She has actively engaged in numerous collaborations with fellow machine learning researchers within the department, aiming to broaden and enhance her expertise in various research domains. She can be contacted at email: mc23018972@studentmail.biust.ac.bw.

Venu Madhav Kuthadi is an Associate professor in the Department of Computing and Informatics at Botswana International University of Science and Technology (BIUST) with research interests in the internet of things, intrusion detection, distributed denial of service, wireless sensor networks, and machine learning. He is currently supervising PhD and MSc students in the areas of internet of things, wireless sensor networks, and machine learning. He is involved in several research projects at BIUST. He can be contacted at email: kuthadiv@biust.ac.bw.

Blessing Omogbehin is an MSc student in the Department of Computing and Informatics at Botswana International University of Science and Technology, Palapye, Botswana focusing on applying machine learning models to enhance cybersecurity. Her research centers on improving data security and access control in cloud computing environments. With a strong foundation in programming, database management, and information security, she has developed practical tools for network analysis and encryption using Python. Passionate about cybersecurity education, Blessing is dedicated to advancing AI-driven solutions that address real-world security challenges. She can be contacted at email: ob24019134@studentmail.biust.ac.bw.