STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **14**, December 2025, pp 3359–3379.
Published online in International Academic Press (www.IAPress.org)

Diabetes Prediction Based on Ensemble Methods

Jihan Askandar Mosa [1,2,*], Adnan Mohsin Abdulazeez [3]

[1] *Information Technology Management Dept., Technical College of Administration, Duhok Polytechnic University, Duhok, Iraq*
[2] *Information Technology Dept., Shekhan Technical Institute, Duhok Polytechnic University, Duhok, Iraq*
[3] *Technical College of Engineering, Duhok Polytechnic University, Duhok, Iraq*

Abstract Diabetes is a chronic disease with rapidly increasing prevalence worldwide, particularly in low- and middle-income countries. Accurate early prediction is essential to reduce complications and improve patient outcomes. This study presents a comprehensive ensemble-based machine learning framework for diabetes prediction using two benchmark datasets: the Pima Indian Diabetes Dataset and the Diabetes Prediction Dataset. A robust preprocessing pipeline was designed, including KNN imputation for missing values, IQR-based outlier removal, SMOTEENN for class balancing, standardization, and forward feature selection to enhance model reliability and interpretability. Two ensemble strategies were implemented and compared: a parallel soft-voting model integrating Logistic Regression, Decision Tree, and K-Nearest Neighbors, and a sequential multi-stage boosting model combining XGBoost, Gradient Boosting, and AdaBoost. Models were evaluated using both a 70%-15%-15% training/validation/testing split and 10-fold cross-validation with metrics such as accuracy, F1-score, precision, recall, and ROC-AUC. Experimental results demonstrated that the sequential ensemble consistently outperformed the parallel model, particularly on the Pima dataset (validation accuracy 97.59%, F1 97.77%, cross-validated accuracy 95.07 ±3.48%) and maintained slightly higher F1 and ROC-AUC on the larger Diabetes Prediction dataset (F1 98.51%, ROC-AUC 99.9%±0.0%). Statistical tests (paired t-test and McNemar's test) confirmed the robustness of these findings. The results show that integrating SMOTEENN balancing, forward feature selection, and boosting-based ensemble learning yields a powerful and generalizable predictive framework suitable for clinical decision support systems in early diabetes diagnosis.

Keywords Diabetes Prediction, Ensemble Learning, Gradient Boosting, AdaBoost,XGBoost

AMS 2010 subject classifications 62P10, 68T05, 62H30

DOI: 10.19139/soic-2310-5070-2771

1. Introduction

Diabetes mellitus [1] is a widespread chronic disease and one of the most serious global health concerns. This occurs as a result of the body's inability to produce or use insulin effectively, resulting in high blood sugar levels and numerous health complications. Middle-aged and older individuals are traditionally the most affected groups [2, 3, 4, 5]. According to recent statistics, 537 million people were diagnosed with diabetes in 2021, with 81% living in low- and middle-income countries. The number of diabetes-related deaths reached 6.7 million, and the diabetic population is projected to increase by 48% by 2045 [6, 7, 8]. Despite the disease's prevalence, an estimated 30% to 50% of individuals with diabetes remain undiagnosed, highlighting the urgent need for early detection and preventive care [9].There are three primary types of diabetes: type 1, type 2, and gestational diabetes mellitus (GDM). Type 1 diabetes, more common among children, occurs when the immune system attacks insulin-producing cells. Type 2 diabetes, prevalent in adults and older individuals, results from insulin resistance.

[*] Correspondence to: Jihan Askandar Mosa (Email: jihan.mosa@dpu.edu.krd) Information Technology Dept., Shekhan Technical Institute, Duhok Polytechnic University, Duhok, Iraq

ISSN 2310-5070 (online) ISSN 2311-004X (print)

GDM is a form of glucose intolerance diagnosed during pregnancy that can cause temporary or long-term health complications for both the mother and child [10, 11, 12, 13, 14].

Early and accurate prediction of diabetes is essential for preventing complications and enabling timely treatment. Recently, Machine Learning (ML) techniques have shown great promise in building predictive models for various chronic diseases. Among these, Ensemble Learning (EL) has proven particularly effective by combining multiple base learners to increase prediction accuracy and model robustness[15, 16]. Ensemble methods can be broadly categorized into two types: parallel and sequential. In parallel ensemble models, base learners are trained independently and their predictions are aggregated (e.g., by majority voting). In sequential ensembles, such as AdaBoost, Gradient Boosting, and XGBoost, base learners are trained in sequence, with each model focusing on correcting the mistakes of the previous one [17, 18]. While ensemble learning is widely used, few studies have directly compared parallel and sequential ensemble methods in the context of diabetes prediction, particularly using multiple datasets and consistent preprocessing pipelines. This study aims to develop a comprehensive machine learning framework for diabetes prediction using both parallel and sequential ensemble strategies. Specifically, we evaluate the performance of a parallel ensemble using a soft Voting Classifier composed of Logistic Regression, Decision Tree, and K-Nearest Neighbors, and a sequential ensemble integrating XGBoost, Gradient Boosting, and AdaBoost in a multi-stage structure. The main contributions of this study are as follows:

- Development of a hybrid ensemble prediction framework that integrates both parallel and sequential ensemble strategies to improve diabetes prediction performance across diverse datasets.
- Implementation of a robust preprocessing pipeline, including K-Nearest Neighbors (KNN) imputation, Interquartile Range (IQR) based outlier removal, Synthetic Minority Over-sampling Technique with Edited Nearest Neighbors (SMOTEENN) for class balancing, and standardization, to ensure data quality and consistency.
- Integration of forward feature selection to enhance model interpretability, reduce dimensionality, and eliminate irrelevant features.
- Design of a sequential ensemble approach, combining XGBoost, Gradient Boosting, and AdaBoost in a multi-stage structure to progressively correct prediction errors and improve generalization.
- Comprehensive evaluation on two real-world medical datasets the Diabetes Prediction Dataset and Pima Indian Diabetes Dataset demonstrating the model's adaptability to different data distributions.
- Comparative analysis between ensemble strategies, highlighting the advantages of hybrid sequential boosting over traditional parallel methods in terms of precision, recall, and area under the curve (ROC-AUC), F1-score, validation accuracy, and training accuracy

Figure 1. Diabetes Managment [19].

This paper's remaining sections are arranged as follows: Section 2 reviews recent research related to diabetes prediction, with a focus on ensemble learning methods and their impact on model performance. Section 3 provides details about the research. Section 4 describes the training procedures of ensemble models and outlines the experimental setup. In Section 5, the performance of the ensemble approaches is compared and the results of applying various feature selection procedures are presented. Section 6 brings the study to a close and offers suggestions for future fields of inquiry in medical prediction systems.

2. Related Work

This section begins by examining studies conducted by current researchers who have applied machine learning techniques to the healthcare system, such as its role in identifying chronic diseases, such as diabetes. In recent years, there has been growing interest in applying ensemble learning techniques to enhance the accuracy of diabetes predictive models. These techniques offer the ability to combine the strengths of multiple algorithms and mitigate their individual weaknesses. A review of the literature identifies previous research efforts into three main categories: stacking, voting, and boosting. Despite the notable successes of these studies, a critical analysis reveals clear gaps in model integration, parameter tuning, and generalizability, which this research seeks to address.

The stacking approach is based on combining multiple base models using a meta-learner to capture integrated patterns and improve predictive performance. For example, Reza et al. (2024) developed a model based on deep neural networks combined with traditional algorithms such as SVMs, decision forests, and decision trees, using logistic regression as the meta-learner. Their model demonstrated an accuracy of 96.91% on real hospital data, 77.10% on PIMA data, and 95.50% on simulated data, demonstrating its strong generalization across diverse data environments. However, the observed decline in performance on PIMA data highlights the sensitivity of this approach to data distribution and sample size, which may limit its practical applicability in various contexts [20]. Similarly, Oliullah et al. (2024) presented a robust clustering approach that incorporates six base learners (XGBoost, NGBoost, AdaBoost, LightGBM, Random Forest, and Bagging) with Bagging as the overlearner, achieving an accuracy of 92.9% after implementing advanced feature engineering techniques [21]. This sudy demonstrates the potential for significant improvement in predictive performance when combining diverse, robust models within a structured sequential framework. However, this type of model requires significant computational resources, precise parameter tuning, and strict procedures to prevent data leakage during cross-validation, which may complicate its reapplication in practical medical settings or resource-constrained systems. These studies reveal that sequential clustering has a high potential for exploiting diversity between models and improving accuracy. However, its high computational cost, sensitivity to data partitioning flaws, and difficulty in tuning may limit its practical application.

Voting (both hard and soft voting) is one of the most popular fusion techniques due to its simplicity and feasibility. It combines the results of multiple learners, either according to the majority rule or via probability averaging. Rashid et al. (2024) implemented a soft voting model combining XGBoost, Random Forest, KNN, logistic regression, and decision trees, along with advanced processing steps including imputation, normalization, and outlier handling. They achieved an accuracy of 81% with high sensitivity and specificity compared to individual models [22]. Ashisha et al. (2024) also focused on the problem of class imbalance using random oversampling and feature selection using the Boruta algorithm. They combined this with a voting ensemble of Random Forest, Gradient Boosting, and LightGBM, achieving an accuracy of 90% on German data and 93% on PIMA data [23]. Kibria et al. (2022) also achieved excellent results, achieving an accuracy balance of 90% and an F1 score of 89% using a smooth voting approach combining Random Forest, XGBoost, AdaBoost, SVM, logistic regression, and artificial neural networks, while employing advanced interpretability tools such as SHAP to increase transparency [24]. Mushtaq et al. (2022) adopted a two-stage approach, first applying SMOTE to balance classes and then combining Random Forest with Gradient Boosting and Naive Bayes, improving accuracy from 80.7% to 81.7% [25]. This approach is simple, computationally demanding, and more stable than sequential pooling or boosting. However, the final performance is highly dependent on the quality and diversity of the underlying models. If the models are similarly biased or have limited diversity, the voting gains are limited, which may hinder its application to problems requiring significant variation in analysis angles.

Boosting relies on sequentially training models, with each model correcting the errors of the previous one, giving it a superior ability to handle complex and imbalanced data. Mazhar et al. (2024) compared several algorithms, including SVM, KNN, logistic regression, XGBoost, and LightGBM, and found that XGBoost achieved the highest accuracy of 90% [26]. Kawarkhe et al. (2024) combined Random Forest, Gradient Boosting, Logistic Regression, LDA, and CatBoost algorithms into voting and stacking models after applying normalization and outlier removal. They achieved an accuracy of 90.62% with high recall and a significant area under the curve (AUC) [27]. Aziz et al. (2024) tested 12 classifiers and found that Random Forest achieved 83% accuracy on its own, then improved performance to 86% through custom fusion using the Fusion method [28]. More recent studies, such as Shao et al. (2024), focused on parameter tuning and data balance. They combined SMOTE with Random Under Sampling and used Optuna to tune LightGBM parameters, resulting in a slight increase in accuracy (from 97.07% to 97.11%) while significantly reducing training time [29]. Kaliappan et al. (2024) also highlighted the importance of feature selection using filtering, wrapping, and interpretive tools such as SHAP and LIME, which helped raise accuracy to 90% and improve interpretability [30]. Although boosting is powerful in improving prediction accuracy and efficiently handles imbalanced data, it can be computationally more expensive than other methods and is prone to overfitting if not carefully tuned or when using small datasets.

This critical analysis demonstrates that while stacking, voting, and boosting methods improve diabetes prediction, each has key drawbacks: stacking is powerful but computationally complex and sensitive to data partitioning and parameter tuning; voting is simple and stable but limited by model diversity; and boosting excels on complex or imbalanced data but requires high computational resources and carries the risk of overfitting. Most recent studies focus on boosting (e.g., XGBoost, Gradient Boosting, and AdaBoost) with preprocessing such as SMOTE, Boruta, or Optuna, or integrating underlying models (decision trees, logistic regression, KNN) into voting. However, they typically focus on a single ensemble type without directly comparing parallel and sequential frameworks within a unified pipeline.

To address this gap, this paper proposes a dual framework that combines parallel soft voting (logistic regression, decision tree, and KNN) and sequential boosting (XGBoost, gradient boosting, and AdaBoost), integrates forward feature selection with splitting (70% training, 15% validation, and 15% testing), and 10-fold cross-validation to ensure the accuracy and generalizability of the results.

3. Research Methodology

This section outlines the methods and tools applied to handle data imbalance and improve classification accuracy across the Diabetes Prediction Dataset. As illustrated in Figure 2, the ensemble learning framework integrates clinical attributes such as glucose level, BMI, insulin, age, and family history to enable robust prediction. This study utilized two widely used tabular datasets: the Diabetes Prediction Dataset and Pima Indian Diabetes Dataset these datasets provide a range of clinical and physiological features.

3.1. Datasets for diabetes

The Diabetes Prediction Dataset and Pima Indian Diabetes Dataset are two publicly accessible datasets that are used in this study to build and assess machine learning models for diabetes prediction. Nine features, including gender, age, hypertension, heart disease, smoking history, BMI, HbA1c level, blood glucose level, and a binary variable indicating the presence or absence of diabetes, are included in the 100,000 records that make up the Diabetes Prediction Dataset. It is ideal for training machine learning models that need multidimensional inputs because of its size and wide range of features. In contrast, While the Pima Indian Diabetes Dataset only contains 768 records, it offers nine clinical features: age, blood pressure, skin thickness, insulin, BMI, number of pregnancies, glucose level, diabetes pedigree function, and a binary outcome that indicates diabetes status. It is well-structured medically and, although it is small in size, it is one of the most popular benchmark datasets for assessing diabetes prediction algorithms. endDatasets for diabetes

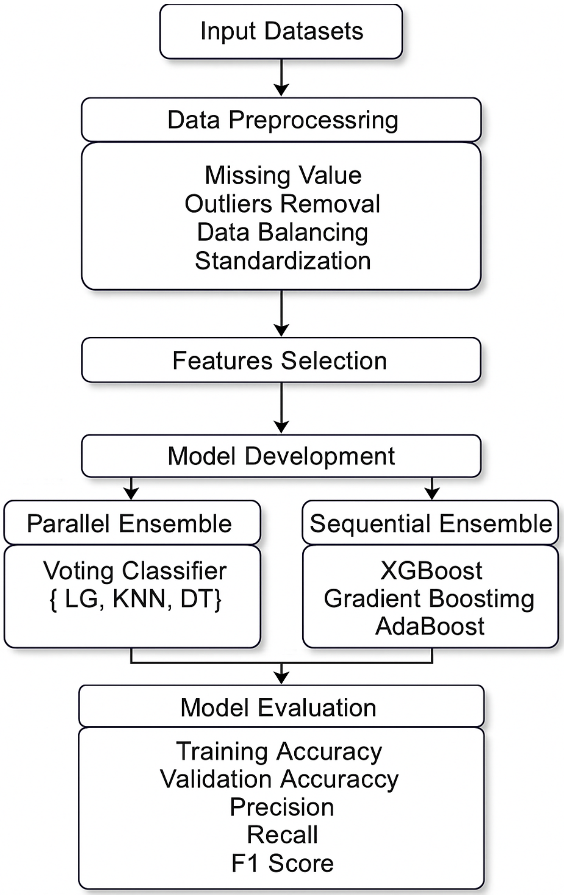


Figure 2. Complete Work Flow of research methodology

3.2. *Data Preprocessing*

Data preprocessing is one of the most critical factors in building successful machine learning models, especially for medical datasets, which often have outliers, missing values, imbalanced distributions, and different feature scales. The present study employed a systematic preprocessing pipeline to ensure quality inputs and enhance model performance.

3.2.1. *Missing Value Imputation*

Both datasets used in this study contain nine features in total, but they are not comprehensive. There are missing values in a number of features in the Pima Indian Diabetes Dataset to varied degrees. For example, *Pregnancies* has missing entries, while *Glucose* and *Blood Pressure* have a few missing values. More significant gaps are observed in *Skin Thickness* and *Insulin*. In contrast, features such as *BMI*, *DiabetesPedigreeFunction*, *Age*, and *Outcome* are complete and contain no missing data, as shown in Figure 3.

Similarly, the Diabetes Prediction Dataset includes nine features, some of which are fully complete while others exhibit substantial missingness. Features such as *gender*, *age*, *smoking_history*, *BMI*, *HbA1c_level*, *blood_glucose_level*, and *diabetes* are fully populated and thus considered highly reliable for analysis. However, *hypertension* and *heart_disease* contain considerable amounts of missing data, necessitating careful preprocessing, as shown in Figure 4.

To address these inconsistencies and ensure high data quality, the *K-Nearest Neighbors (KNN)* imputation method was employed. This method estimates missing values by identifying the k most similar records (based on Euclidean distance) and imputing missing entries with the average values from these neighbors. KNN imputation is particularly suitable for medical datasets because it preserves the natural relationships among features and maintains the overall data distribution. This step contributed to generating reliable training data and enhancing the overall performance and robustness of the predictive model.

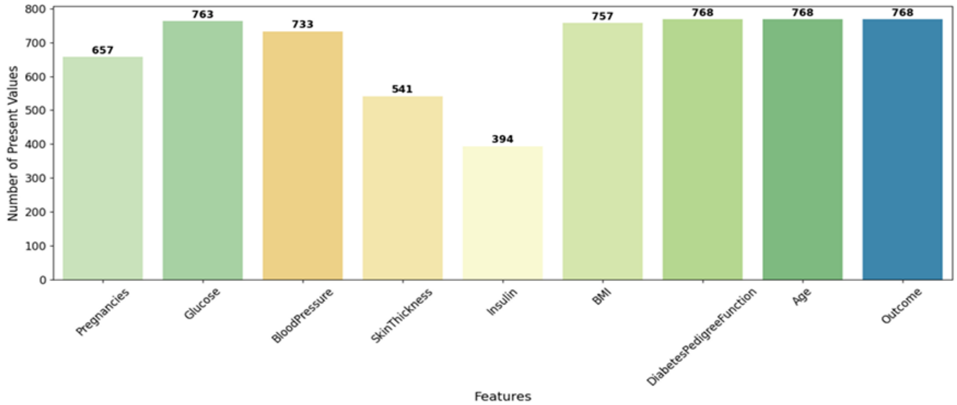


Figure 3. Each feature’s number of missing features in the dataset (Pima Indian dataset).

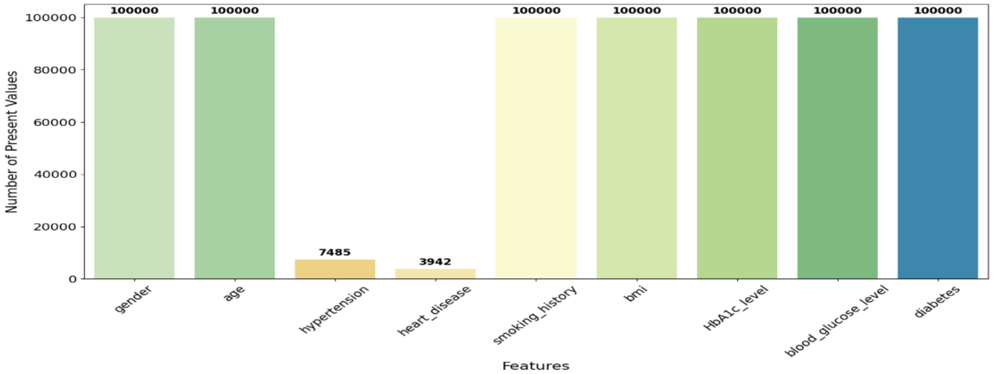


Figure 4. Each feature’s number of missing features in the dataset (Diabetes prediction dataset)

3.2.2. *Outlier Removal*

Outlier scores were derived and computed from the dataset using the *Interquartile Range (IQR)* outlier method. This method effectively removes extreme values that might otherwise adversely influence the model and compromise its predictive ability. Eliminating these outliers enhances the quality of the data and contributes to establishing a more stable and reliable predictive model.

3.2.3. *Data Balancing*

Balancing the dataset is a crucial aspect of preprocessing, especially for medical data such as the *Diabetes Prediction Dataset* and the *Pima Indian Diabetes Dataset*, which typically exhibit significant class imbalance.

Such imbalance often leads to biased models that achieve high accuracy on the majority class but perform poorly on the minority class.

To address this problem, the *Synthetic Minority Over-sampling Technique with Edited Nearest Neighbors (SMOTEENN)* was applied. SMOTEENN combines two operations:

- **SMOTE**: Oversamples minority-class cases by generating synthetic samples.
- **ENN**: Cleans the dataset by removing noisy or incorrectly classified instances from the majority class.

This hybrid process produces a cleaner, more balanced dataset, improving model performance and stability.

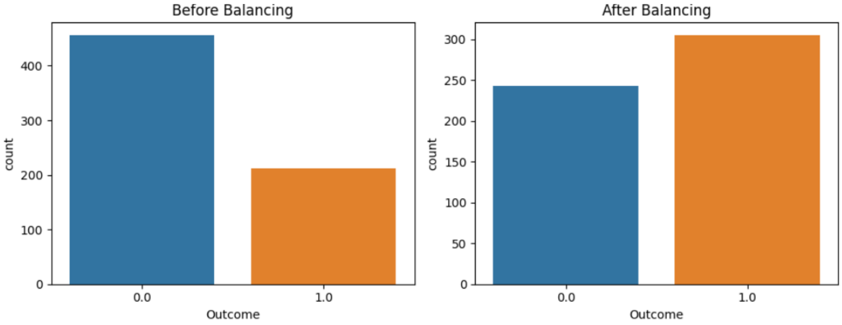


Figure 5. Data Distribution Before and After Balancing (Pima Indian dataset).

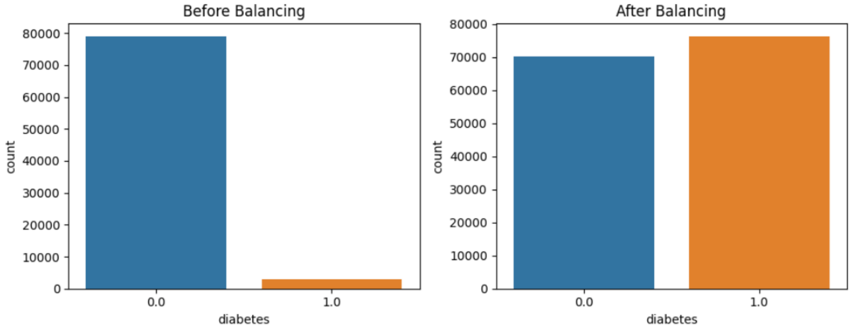


Figure 6. Data Distribution Before and After Balancing (Diabetes prediction dataset)

Figure 5 illustrates the data distribution of the Diabetes Prediction Dataset before and after SMOTEENN application. The left bar chart shows the original unbalanced data, while the right chart shows the much more balanced distribution achieved by this method. Similarly, Figure 6 presents the Pima Indian Diabetes Dataset's data distribution before and after applying SMOTEENN. In both cases, the algorithm significantly improved data balance, thereby reducing model bias and increasing overall prediction accuracy.

3.2.4. Standardization

Standardizing means transforming the features so that they appear as having a mean of 0 and unit variance. This ensures that every feature contributes equally to the learning process irrespective of its original scale. It prevents larger features from dictating the fate of the model and becomes even more crucial in the case of features with different ranges. It improves training stability, speeds up convergence, and reduces overfitting to provide more precise and reliable predictions

3.3. Feature Selection Techniques

In machine learning, *feature selection* is the process of identifying and selecting a subset of relevant variables that contribute most to the predictive model. This process reduces data dimensionality, eliminates redundant or irrelevant variables, and ultimately improves model interpretability and performance.

3.3.1. Forward Feature Selection

The *forward selection* technique begins with an empty set of features and iteratively adds one feature at a time based on its contribution to the model's predictive power. This procedure continues until a predefined number of features is reached or no significant improvement is observed. By focusing only on the most influential features, forward selection reduces noise, lowers the risk of overfitting, and enhances model performance.

3.3.2. Implementation of Forward Feature Selection

In this study, forward feature selection was applied independently to both the *Diabetes Prediction Dataset* and the *Pima Indian Diabetes Dataset* to identify the most relevant attributes associated with diabetes risk. The implementation utilized the `SequentialFeatureSelector` module from the scikit-learn library. Starting with no features, the most influential predictors were gradually added based on their incremental contribution to model accuracy.

After selection, the reduced feature sets were used to retrain all ensemble models-both sequential and parallel. The results confirmed that incorporating forward feature selection into the preprocessing pipeline significantly improved training accuracy, validation accuracy, and F1-score, thereby enhancing overall model generalization and robustness.

3.4. Proposed Algorithms

Ensemble methods are advanced machine learning techniques that combine the predictions of multiple base models to improve overall accuracy, stability, and generalization. In this study, both parallel and sequential ensemble approaches were applied. The parallel ensemble is represented by Voting Classifier that aggregates the outputs of three independent models: Logistic Regression, K-Nearest Neighbors, Decision Tree. This method enhances prediction by leveraging the diversity of models working in parallel. On the other hand, sequential ensemble methods including AdaBoost, Gradient Boosting, and XGBoost train models in sequence, where each model learns from the errors of its predecessor. These methods are particularly effective in minimizing prediction errors and boosting overall performance. By comparing both types, this research provides valuable insights into their effectiveness in diabetes prediction.

3.4.1. Parallel Ensemble Models

Parallel ensemble models work by training multiple base learners independently on the same dataset and then combining their predictions. The main idea is that different models may capture different aspects of the data. By aggregating their outputs (e.g., via majority voting), the ensemble can reduce individual errors and improve overall accuracy.

In this study, was implemented to represent the parallel ensemble approach. It combines three diverse classifiers:

- **Logistic Regression** - a linear, interpretable model suitable for binary classification.
- **Decision Tree Classifier** - a rule-based model capable of modeling complex, non-linear relationships.
- **K-Nearest Neighbors (KNN)** - a distance-based model that classifies based on local data distribution.

Each model was trained independently on the same preprocessed dataset. The final prediction was obtained using soft voting, where the most frequent class among the three outputs was selected. For example, if two out of three models predicted ”diabetic”, the final output would be ”diabetic”, as shown in figure 7.

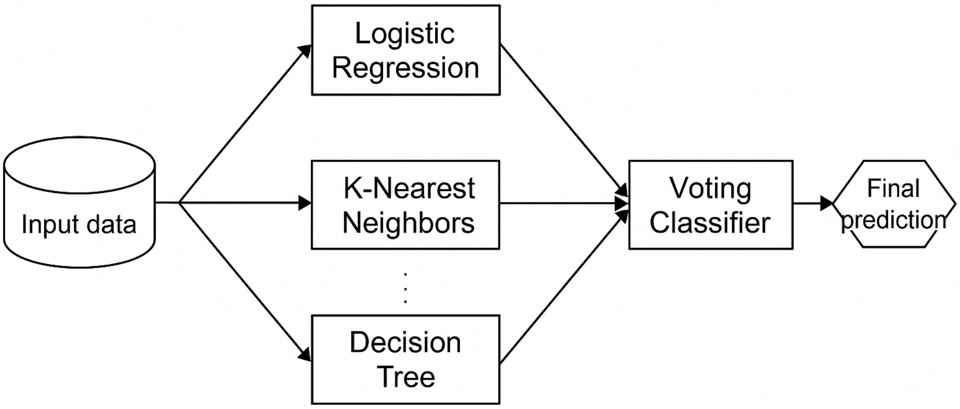


Figure 7. Block diagram illustrating the Parallel Ensemble

3.4.2. *Sequential Ensemble Models*

The sequential ensemble model used in this research is the core innovation, aiming to improve diabetes prediction performance by addressing errors in successive stages. The model consists of three stages:

1. **Stage 1**: An XGBoost model is trained on the entire training set and misclassified samples are identified.
2. **Stage 2**: A Gradient Boosting model is trained only on these misclassified samples to correct XGBoost errors.
3. **Stage 3**: An AdaBoost model is applied to the samples misclassified by Gradient Boosting, aiming to enhance the accuracy of the final predictions.

After completing the three stages of the sequential model, the final prediction is reached through a clear and organized mechanism. Initially, the XGBoost model generates its outputs, which represent the first line of defense. If these outputs are accurate and highly confident, they are adopted immediately. In cases where classification errors or uncertainties appear, these samples are transferred to the Gradient Boosting model, which works to address the remaining errors. If difficult samples remain that have not been classified correctly, the AdaBoost model intervenes in the final stage to provide the final decision. This gradual sequence ensures that each stage adds additional corrective value, reducing errors step by step until we reach a final prediction that is more accurate and reliable than any single model, as illustrated in Figure 8.

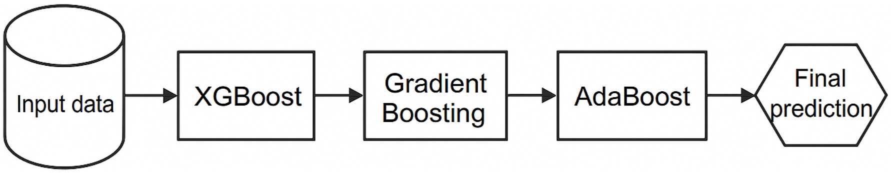


Figure 8. Block diagram illustrating the sequential ensemble

Extreme Gradient Boosting (XGBoost). XGBoost is an advanced version of gradient boosting that strategically combines sequential learning at the model-building level with parallel execution of computations within each boosting iteration. While trees are added one at a time in a sequential manner each new tree focusing on correcting the errors of its predecessor XGBoost optimizes this process by parallelizing operations such as finding the best split points and evaluating gradients. It incorporates second-order derivatives in its loss function, allowing for more accurate optimization.

Gradient Boosting. Gradient Boosting is a sequential ensemble learning method where models are trained one after another in a step-by-step manner. This inherently sequential approach means that each new weak learner is trained to correct the residual errors made by the previous ensemble. Unlike Random Forest, where trees are built independently, Gradient Boosting adds trees in a series, each one focusing on the mistakes of the ensemble so far. While this design enhances the ability to minimize bias and fit complex, non-linear relationships, it also means that training is computationally more demanding and harder to parallelize at the model level.

Adaptive Boosting (AdaBoost). AdaBoost is a classic ensemble method that builds a strong classifier by sequentially combining multiple weak learners typically simple decision stumps. The algorithm is fundamentally sequential, as each new learner is trained based on the performance of the previous ones. Initially, all training samples are given equal weights, but after each iteration, the weights of misclassified samples are increased. This adaptive weighting forces the subsequent learners to focus on the harder, misclassified instances, improving the model's ability to capture subtle and complex cases. Each weak learner contributes to the final prediction with a weight based on its accuracy, and the final decision is made through a weighted vote. While the sequential nature of AdaBoost limits opportunities for parallel execution during training, the method remains computationally efficient due to the simplicity of its base learners. AdaBoost is particularly valuable in medical prediction tasks such as diabetes diagnosis, where it excels at enhancing performance by paying close attention to difficult-to-classify cases, leading to improved sensitivity and specificity in risk detection.

3.5. Evaluation Metrics

Evaluation metrics play a fundamental role in measuring the efficiency of machine learning models and determining their effectiveness in accurately classifying cases, especially when dealing with medical datasets containing significant class imbalance. In this study, a set of evaluation metrics was adopted to provide a comprehensive and accurate assessment of the performance of ensemble models. These metrics include the following:

3.5.1. Confusion Matrix

The *confusion matrix* is one of the most prominent tools used to evaluate the performance of classification models. It enables researchers to understand the nature of a model's errors, not just its overall accuracy. The matrix displays the model's prediction results in four main categories:

- **True Positives (TP)**: The number of diabetic cases correctly classified as diabetic.
- **True Negatives (TN)**: The number of non-diabetic cases correctly classified as non-diabetic.
- **False Positives (FP)**: The number of non-diabetic cases incorrectly classified as diabetic.
- **False Negatives (FN)**: The number of diabetic cases incorrectly classified as non-diabetic.

This distribution is typically presented by visualizing the confusion matrix as in Table 1.

Table 1. Confusion Matrix

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

3.5.2. Training Accuracy

Training accuracy reflects the model's ability to correctly classify the data on which it was trained. However, relying solely on this metric can lead to misleading conclusions, especially in the presence of overfitting, where a model performs well on training data but poorly on unseen data.

3.5.3. Validation Accuracy

Validation accuracy measures the percentage of correct predictions made by the model when tested on previously unseen validation data. It is a key measure of the model's ability to generalize and perform well on new, real-world data.

3.5.4. Precision

Precision quantifies the proportion of positive predictions that are actually correct, providing insight into the model's accuracy when it predicts a positive case:

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

3.5.5. Recall (Sensitivity)

Recall, also known as sensitivity, measures the model's ability to detect true positive cases within the dataset:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

3.5.6. F1-Score

The F1-score is the harmonic mean of precision and recall, providing a balanced measure of both metrics. It is particularly effective and reliable in cases of class imbalance:

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

3.5.7. ROC-AUC

The *Receiver Operating Characteristic - Area Under the Curve (ROC-AUC)* metric evaluates the performance of binary classification models. The ROC curve plots the relationship between the true positive rate (sensitivity) and the false positive rate across all possible decision thresholds. The AUC represents the area under this curve, summarizing the model's ability to distinguish between positive and negative classes. A higher AUC value indicates stronger model performance.

3.5.8. Processing Time

Although not considered a primary success criterion, processing time was also recorded. It provides important insights into the computational efficiency of the proposed ensemble methods, which is particularly relevant when comparing models intended for real-time or large-scale medical applications.

4. Experimental Setup

All models were implemented in Python within a Jupyter Notebook environment. Two evaluation schemes were used to ensure robust and reliable assessment:

(i) Holdout split (70/15/15). Data were split into 70% training, 15% validation, and 15% testing to monitor model performance and control overfitting.

(ii) Stratified 10-fold cross-validation. The dataset was randomly partitioned into ten equally sized folds; in each iteration, nine folds were used for training and one for testing, with each sample serving exactly once as test data. After ten iterations, the mean and standard deviation of all metrics were computed: Accuracy, F1-score, Precision, Recall, and ROC-AUC.

This study relied on ensemble learning to improve prediction accuracy by combining multiple models and reducing errors. Parallel ensembles (Logistic Regression, Decision Tree, and K-Nearest Neighbors via a Voting Classifier) train models independently and aggregate predictions. Sequential ensembles (AdaBoost, Gradient Boosting, and XGBoost) train models in sequence, with each subsequent model focusing on correcting the errors of the previous model. Using both the 70/15/15 split and 10-fold cross-validation enabled robust training and statistically supported evaluation suitable for medical prediction tasks.

5. Results and Discussion

This section presents experimental results from applying various ensemble machine learning models to selected diabetes datasets. The models were evaluated using key performance metrics, including training accuracy, validation accuracy, F1 score, precision, recall, and confusion matrix analysis. The goal is to identify models that provide the most reliable and generalizable predictions, especially in the presence of class imbalance.

5.1. Results on Pima Indian Diabetes Dataset

Ensemble models such as Voting Classifier (in parallel) and sequential models such as XGBoost, Gradient Boosting, and AdaBoost were applied to the Pima Indian Diabetes dataset. This was followed by preprocessing, imbalance handling using SMOTEENN, and feature selection, where only the top 5 features were retained. The results showed a significant improvement in the performance of the models. The results showed significant improvements in model performance, particularly in validation metrics such as accuracy, precision, recall, and F1 score. Table 2 presents the comparative performance of the parallel and sequential ensemble models on the preprocessed dataset.

Table 2. Parallel vs. sequential ensembles on Pima (holdout 70/15/15).

Model	Train Acc	Val Acc	Precision	Recall	F1
Parallel Ensemble	98.16	96.38	100	93.47	96.62
Sequential Ensemble	98.95	97.59	100	0.9565	.97.77

Based on the results of Table 3, which displays the 10-fold cross-validation evaluation of the Pima Indian Diabetes Dataset, the sequential ensemble model outperformed the parallel ensemble model across all key performance indicators. The sequential model achieved an accuracy of 95.07 ±3.48 compared to 91.79 ±4.33 for the parallel model, with a higher F1 score of 95.63 ±3.07 compared to 92.82 ±3.62, reflecting a better balance between precision and recall. It also outperformed in precision (95.06 ±4.94) and recall (96.39 ±3.13), demonstrating a greater ability to detect infected cases and reduce false positives. Additionally, the sequential model achieved a higher area under the curve (ROC-AUC = 94.92 ±3.71) compared to 91.48 ±4.70, confirming its superior ability to distinguish between positive and negative cases at various decision thresholds. Statistical analyses supported these results; the paired t-test revealed a statistically significant difference in the performance of the F1 measure (p = 0.0286 ¡ 0.05), while the McNemar test showed that the error pattern did not differ significantly between the two models (p = 0.25). These results reflect the superiority of the sequential approach based on XGBoost, Gradient Boosting, and AdaBoost algorithms, which relies on progressive error correction and balance mechanisms (SMOTEENN), enhancing its generalization ability and providing higher accuracy and stability, making it a more reliable choice for medical applications that require high sensitivity and accuracy in diabetes diagnosis.

Table 3. 10-fold CV (mean ± std) on Pima Indian Diabetes Dataset (values in %).

Model	**Accuracy (%)**	**F1 (%)**	**Precision (%)**	**Recall (%)**	**ROC-AUC (%)**
Parallel Ensemble	91.79 ± 4.33	92.82 ± 3.62	91.48 ± 6.06	94.42 ± 2.54	91.48 ± 4.70
Sequential Ensemble	95.07 ± 3.48	95.63 ± 3.07	95.06 ± 4.94	96.39 ± 3.13	94.92 ± 3.71

Figure (9) shows a slight superiority of the sequential model over the parallel model in terms of training and validation accuracy. The sequential model achieved a training accuracy of 98.95% and a validation accuracy of 97.59%, compared to 98.16% and 96.38%, respectively, for the parallel model. These results indicate that the sequential model is capable of learning from data without overfitting, with a high ability to generalize to new data. Figure (10), which displays the F1 score, reflects the model's ability to balance precision and recall. The sequential model achieved a higher F1 (97.77%) compared to the parallel model (96.62%), demonstrating its superior ability to handle both positive and negative categories more evenly, a requirement in the medical field to reduce diagnostic errors.

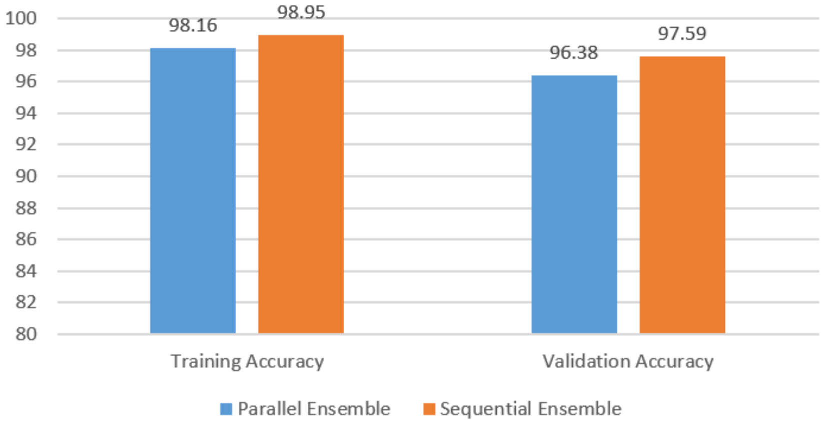


Figure 9. Comparison of Ensemble Models Training and Validation Accuracy

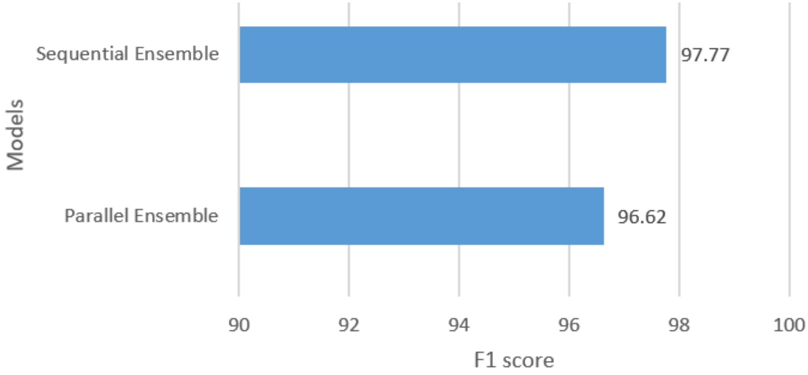
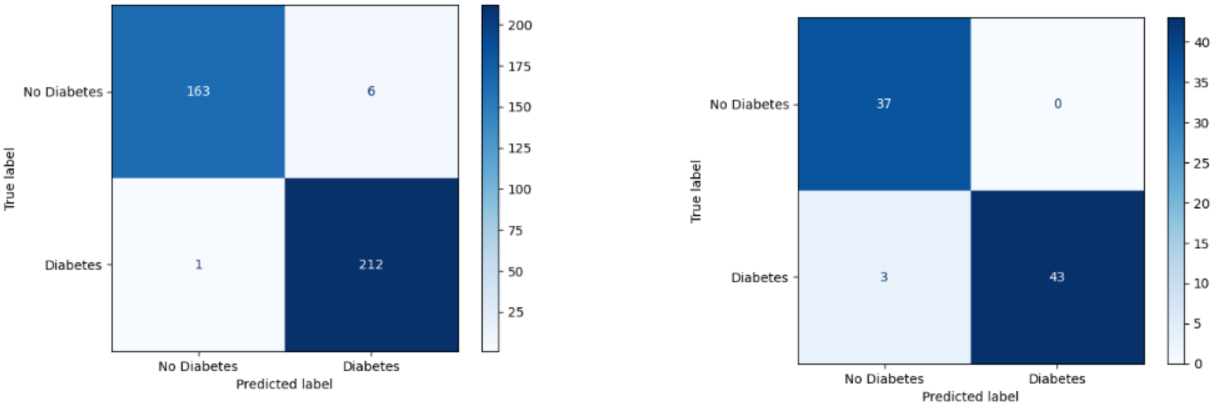


Figure 10. F1 score comparison Parallel vs Sequential Ensemble

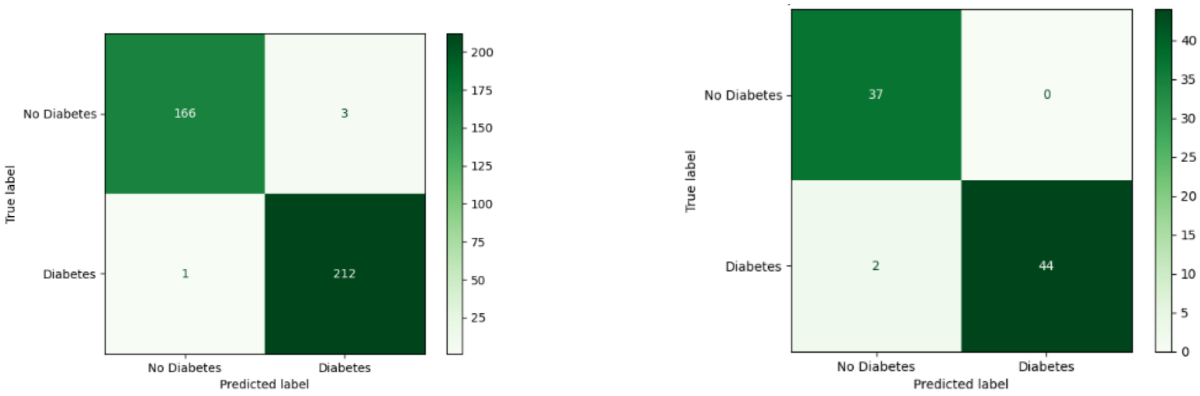
Referring to Figures 11 and 12, which display the confusion matrices, a clear difference can be observed in each model's ability to classify infected cases. Figure 11 for the parallel model shows a larger number of cases that were not detected as having diabetes (false negatives), which is reflected in a lower recall rate (93.47%). Figure 12 for the sequential model shows a significant improvement in this aspect, with recall reaching 95.65%, demonstrating the model's ability to correctly detect a larger number of infected cases.



(a) Training Accuracy confusion Matrix

(b) Validation Accuracy confusion Matrix

Figure 11. In figure (a) and (b) show confusion matrices for Parallel Ensemble (Pima Indian Dataset).



(a) Training Accuracy confusion Matrix

(b) Validation Accuracy confusion Matrix

Figure 12. In figure (a) and (b) show confusion matrices for Sequential Ensemble (Pima Indian Dataset).

Based on these results, it can be argued that the sequential clustering model has higher prediction efficiency, especially in detecting the positive class (infected cases), which is an important criterion in healthcare applications that require the highest levels of accuracy and sensitivity. Its stability in performance across multiple metrics (Accuracy, F1, Recall) also makes it more reliable compared to the parallel model.

5.2. Results on Diabetes Prediction Dataset

Ensemble models, including Voting Classifier (as a parallel clustering model), as well as sequential models such as XGBoost, Gradient Boosting, and AdaBoost, were applied to the Pima Indian Diabetes dataset. Before training the models, the data underwent several basic preprocessing steps, including class imbalance resolution using

SMOTEENN and feature selection, where only the top six relevant features were retained.These improvements resulted in significant improvements in the model performance, particularly across validation metrics such as accuracy, precision, recall, and F1 score. Table 4 summarizes the performance comparison between the parallel and sequential clustering models after applying all the preprocessing steps.

Table 4. Parallel vs. sequential ensembles on Diabetes Prediction (holdout 70/15/15).

Model	**Train Acc**	**Val Acc**	**Precision**	**Recall**	**F1**
Parallel Ensemble	99.36	98.19	97.22	98.36	98.28
Sequential Ensemble	98.47	98.45	98.96	98.05	98.51

Based on the results of Table 5, which displays the 10-Fold Cross-Validation evaluation of the Diabetes Prediction Dataset, it is clear that both the Voting/Parallel Ensemble and Sequential Ensemble models provide very high performance, but some indicators reveal subtle differences that have practical and statistical significance. The parallel model achieved an accuracy of 98.3 ±0.1 and an F1-score of 98.4 ±0.1 with a precision of 97.5 ±0.2 and a high recall of 99.4 ±0.1, while the serial model achieved an accuracy of 98.4 ±0.1 and an F1-score of 98.5 ±0.1 with a higher precision of 98.8 ±0.1 and a slightly lower recall of 98.2 ±01, in addition to a slight superiority in area under the curve (ROC-AUC = 99.9 ±0.0) compared to 99.8 ±0.0. A paired t-test on the F1 measure (t = -1.2042, p = 0.2592) shows that these differences in means are not statistically significant, meaning that the performance of the two models is very similar in terms of the balance between precision and recall. However, the McNemar test revealed a highly statistically significant difference (statistic = 979.5952, p ¡ 0.0001) in the error pattern between the two models, indicating that the types of examples each model misses differ significantly. Therefore, it can be argued that the sequential model has a better ability to distinguish precise boundaries between classes (reflected in precision and ROC-AUC), while the parallel model maintains a slight advantage in recall, a crucial aspect for reducing false negatives in clinical applications. These results reflect that the choice of the optimal model depends on the priority of the clinical goal: if minimizing false negatives is the primary goal, the parallel model is a strong choice, whereas if the need to increase positive precision and overall discrimination is higher, the sequential model is more appropriate.

Table 5. 10-fold CV (mean ± std) on Diabetes Prediction Dataset (values in %).

Model	**Accuracy (%)**	**F1 (%)**	**Precision (%)**	**Recall (%)**	**ROC-AUC (%)**
Parallel Ensemble	98.3 ± 0.1	98.4 ± 0.1	97.5 ± 0.2	99.4 ± 0.1	99.8 ± 0.0
Sequential Ensemble	98.4 ± 0.1	98.5 ± 0.1	98.8 ± 0.1	98.2 ± 0.1	99.9 ± 0.0

Figure 13 shows the training and validation accuracies for both models. It shows that the parallel ensemble model achieved higher training accuracy (99.36%) compared to the sequential model (98.47%), but the sequential model outperformed in validation accuracy (98.45% versus 98.19%), indicating better generalization and less overlearning. These results suggest that the sequential model better avoids overlearning and maintains robust performance when dealing with new data, which is critical in clinical applications.Figure 14 shows a comparison of the F1 scores of the two models. The sequential model achieved a higher F1 score (98.51%) compared to 98.28% for the parallel model, reflecting a better balance between precision and recall, which is critical when prediction errors (false positives or negatives) are costly. Given its superior F1 score, the sequential model is the preferred choice in predictive healthcare systems, where a delicate balance between correct detection and error reduction is required.

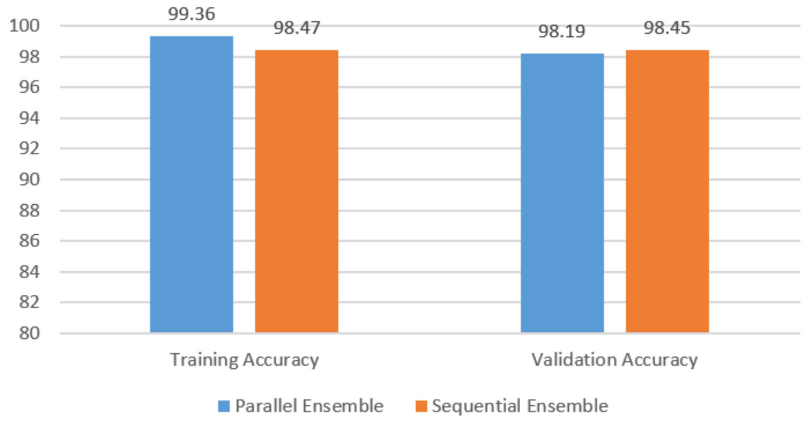


Figure 13. Comparison of Ensemble Models Training and Validation Accuracy

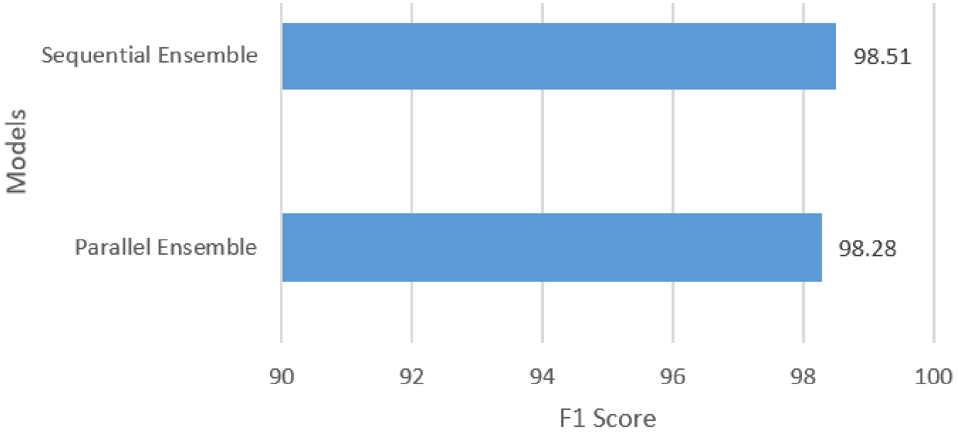
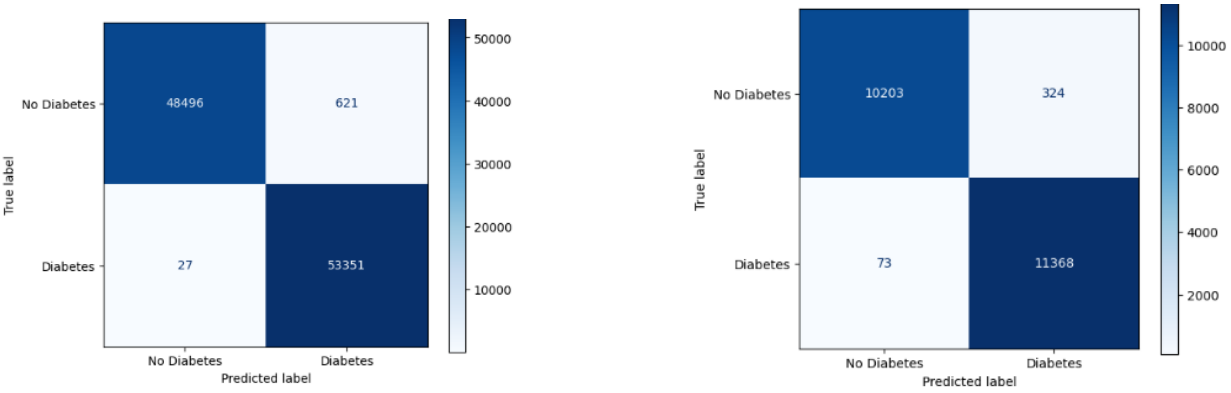


Figure 14. F1 score comparison Parallel vs Sequential Ensemble

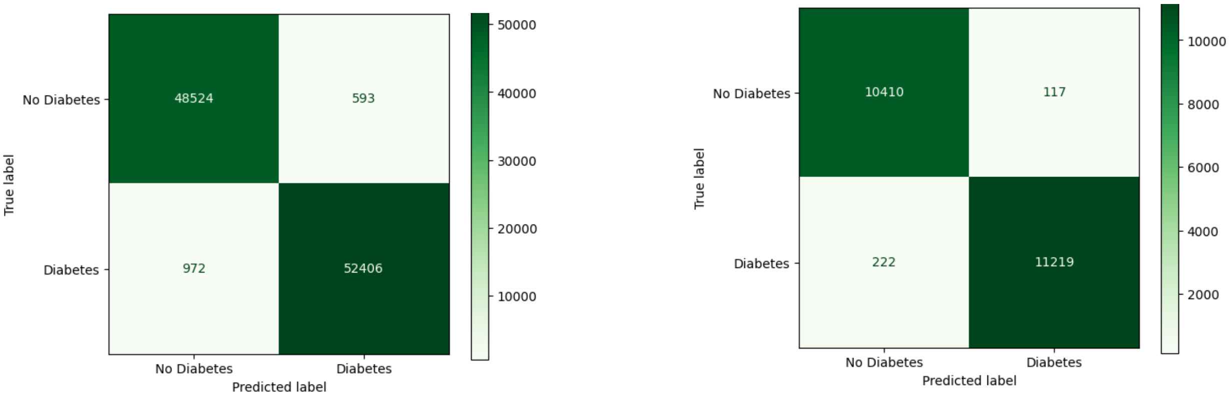
Figures 15 and 16 show the confusion matrices for the two models. Figure 15 (for the parallel model) shows good classification ability, but with a relatively higher number of false negatives compared to the sequential model. Figure 16 (for the sequential model) shows a more balanced matrix, with a clear ability to detect positive (infected) cases and reduce errors. The sequential structure allows the model to gradually learn from previous errors, improving its sensitivity to the positive class and reducing misses, which is critical for diabetes prediction.



(a) Training Accuracy confusion Matrix

(b) Validation Accuracy confusion Matrix

Figure 15. In figure (a) and (b) show confusion matrices for Parallel Ensemble (Diabetes Prediction Dataset).



(a) Training Accuracy confusion Matrix

(b) Validation Accuracy confusion Matrix

Figure 16. In figure (a) and (b) show confusion matrices for Sequential Ensemble (Diabetes Prediction Dataset).

Based on the results, the sequential clustering model demonstrates superior prediction efficiency, particularly in detecting cases (positive class). This performance is in line with the requirements of healthcare systems that require high levels of recall, precision, and generalizability. The consistent performance across multiple indicators also makes the sequential model a more reliable and effective option compared to the parallel model.

5.3. Comparative Analysis

Figure 17 shows a comparison of the F1 scores achieved by the parallel and sequential Ensemble models across three datasets such as Pima Indian Diabetes Dataset, and Diabetes Prediction Dataset. In all cases, the sequential clustering model performed better on the Pima Indian Diabetes Dataset, with the sequential model achieving 97.77% compared to 96.62% for the parallel model, a difference of more than 1.1%. On the Diabetes Prediction Dataset, the performance of the two models was close, with the sequential model achieving 98.51% compared to 98.28% for the parallel model.

These results confirm the superiority of the sequential clustering model in terms of reliability and predictive efficiency, especially in sensitive medical applications. This superiority is often attributed to the incremental learning mechanism in models like XGBoost ,Gradiant Boosting,and AdaBoost, which helps improve performance by correcting errors during training.

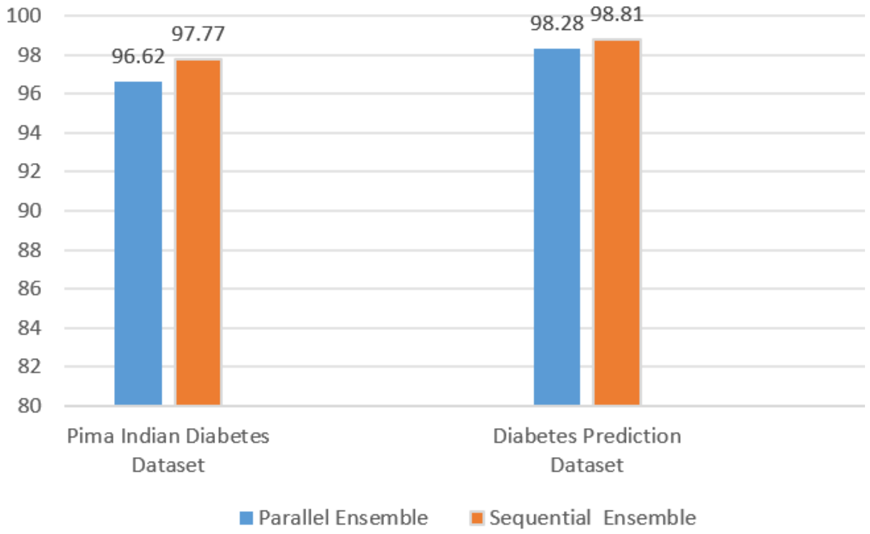


Figure 17. Comparison Results of all Dataset

5.4. Discussion

The results of this study showed that the sequential model clearly outperformed the parallel voting model in both evaluation methods: regular splitting (70% training - 15% validation - 15% testing) and cross-validation (10-fold cross-validation), with differences in the degree of outperformance between the two datasets.When using regular splitting on the Pima Indian Diabetes dataset, the sequential model achieved a training accuracy of 98.95% and a validation accuracy of 97.59%, with an F1 score of 97.77%, outperforming the parallel voting model, which achieved a training accuracy of 98.16%, a validation accuracy of 96.38%, and an F1 score of 96.62%. On the Diabetes Prediction dataset, the sequential model performed at a training accuracy of 98.47%, a validation accuracy of 98.45%, and an F1 of 98.51%, compared to a training accuracy of 99.36%, a validation accuracy of 98.19%, and an F1 of 98.28% for the parallel voting model. These results reflect the sequential model's ability to achieve a better balance between precision and recall while maintaining high generalization ability, even when the data size is larger and more diverse.

When applying 10-fold cross-validation, the sequential model showed greater superiority on the Pima Indian Diabetes dataset, achieving an average accuracy of 95.07 ±3.48, F1 of 95.63 ±3.07, precision of 95.06 ±4.94, recall of 96.39 ±3.13, and a ROC-AUC of 94.92 ±3.71, compared to the parallel voting model, which recorded an accuracy of 91.79 ±4.33, F1 of 92.82 ±3.62, precision of 91.48 ±6.06, recall of 94.42 ±2.54, and ROC-AUC of 91.48 ±3.71. On the Diabetes Prediction dataset, performance was similar, with the sequential model achieving an average accuracy of 98.4 ±0.1, F1 of 98.5 ±0.1, precision of 98.8 ±0.1, recall of 98.2 ±0.1, and ROC-AUC of 99.9 ±0.0 The parallel voting model achieved an accuracy of 98.3 ±0.1, F1 of 98.4 ±0.1, precision of 97.5 ±0.2, recall of 99.4 ±0.1, and ROC-AUC of 99.8 ±0.0.

These results demonstrate that the incremental learning and error correction mechanisms of the sequential model give it a clear ability to improve performance, especially with small or imbalanced datasets such as the Pima set, while maintaining high and balanced performance on larger datasets such as the Diabetes Prediction set. The combination of regular partitioning and cross-validation also demonstrates that the achieved results are not transient or limited to a particular dataset, but rather reflect a true generalization capability, making the proposed framework a reliable choice for medical applications requiring high accuracy and stability in early diabetes prediction.

5.4.1. Comparison to Prior Works

To comprehensively evaluate the proposed dual-clustering framework (parallel voting and sequential lifting), the results were compared with several recent studies that addressed diabetes prediction using clustering techniques or advanced strategies to address data imbalance. Table 6 summarizes these comparisons. These studies relied on various approaches, such as combining stacking with deep networks, using smooth and weighted voting methods, and using fusion and boosting models with data balance techniques. Despite these advances, most of these studies demonstrated lower accuracy and F1 scores on the Pima Indian Diabetes and Diabetes Prediction datasets compared to the proposed sequential model.

Despite these methodological advances, most of these studies demonstrated lower accuracy and F1 scores on both the Pima Indian Diabetes dataset and the Diabetes Prediction dataset compared to the proposed sequential model. For example, the fusion model in Aziz et al. (2024) achieved an accuracy of 86%b [28], Kawarkhe et al. (2024) achieved an accuracy of 90.6% using diverse clustering [27], Oliullah et al. (2023) achieved an accuracy of 92.9% using a stacked model [21], and Kibria et al. (2022) achieved a balanced accuracy of approximately 90% with an F1 value of approximately 89% [24]. Mushtaq et al.'s (2022) study achieved an accuracy of no more than 81.7% using a voting algorithm [25], while Shao et al.'s (2024) study, enhanced with LightGBM and Optuna, achieved an accuracy of 97.1% [29]. Kaliappan et al.'s (2024) study achieved the highest accuracy of 94% on the Diabetes Prediction dataset [30]. In contrast, the proposed framework achieved an accuracy of 98.45% and an F1 value of 98.51% on the Diabetes Prediction dataset, and an accuracy of 97.59% and an F1 value of 97.77% on the Pima Indian Diabetes dataset, with stable results verified using 10-fold cross-validation. This superiority is attributed to the combination of the SMOTEENN data balancing technique with forward feature selection to identify the most important variables, and to the multi-stage error correction mechanism integrated into the chain-up components.

Table 6. Comparison with prior works on diabetes prediction.

Ref #	Year	Dataset	Techniques Used	Results
[28]	2024	PIMA	Fusion (Ensemble): RF, DT, SVC, KNN	RF: 83.0%, Fusion: 86.0%
[27]	2024	PIMA	LR, RF, GBC, LDA, CatBoost	Mixed ensemble: 90.6%
[21]	2023	PIMA	GridSearchCV; XGBoost, NGBoost, Bagging, LightGBM, AdaBoost, RF	Highest Accuracy: 92.9%
[24]	2022	PIMA	Weighted Ensemble: ANN, RF, SVM, LR, AdaBoost, XGBoost	Balanced Accuracy: ∼90%
[25]	2022	PIMA	RF, NB, SVM, KNN, Gradient Boost, LR, Voting Classifier	RF: 80.7%, Voting Classifier: 81.7%
[29]	2024	Diabetes Prediction Dataset	LightGBM	Accuracy: 97.0%–97.1%
[30]	2024	Diabetes Prediction Dataset	RF, XGB, LR, GB, SVM	Accuracy: 94.0%
Ours	(DP): Val Acc. 98.45%, F1 98.51%;	(PIMA): Val Acc. 97.59%, F1 97.77%.		

6. Conclusion and Future Work

This study presented an integrated framework for diabetes prediction using ensemble machine learning techniques. It compared two main models: a parallel ensemble model (Soft Voting), which combines logistic regression, decision tree, and the K-nearest Neighbor (KNN) algorithm, and a sequential ensemble model, which combines XGBoost, Gradient Boosting, and AdaBoost algorithms in a multi-stage structure, with each level correcting for errors from the previous level. The proposed framework included a comprehensive series of preprocessing steps, including: K-nearest Neighbor (KNN) compensation for missing values, IQR outlier removal, SMOTEENN balancing of unequal data, and standardization of ranges using standard scaling. Forward feature selection was also performed to enhance model accuracy, reduce complexity, and improve interpretability.The results showed that the sequential model achieved a clear advantage, particularly on the Pima Indian Diabetes dataset, where the validation accuracy reached 97.59% and the F1 coefficient reached 97.77%, with an average cross-validation accuracy (10-fold CV) of 95.07% ±3.48%. Despite the convergence of results on the larger diabetes prediction dataset, the sequential model maintained a slight advantage in key metrics such as the F1 coefficient (98.51%) and the area under the ROC-AUC (99.9% ±0.0%). However, the parallel model demonstrated limited recall, making it suitable when reducing undetected cases is a top priority.These results are attributed to the combination of data balancing and feature selection techniques with sequential error correction mechanisms in the reinforcement learning model, which increased the generalization ability of the models, reduced the effect of bias resulting from data imbalance, and confirmed their efficiency in sensitive medical applications that require high accuracy and high sensitivity.

In future work, it may be possible to extend this proposed framework to include more sophisticated patterns via deep learning. It could also benefit from deeper hyperparameter optimization through techniques such as Optuna or grid search. Finally, the model could be implemented in a clinical decision support system (CDSS) and refinement through the application of explainability techniques such as SHAP and LIME so as to advance transparency and trustworthiness in real-world healthcare settings.

Data Availability

In this study, examination was performed on datasets which were publicly available. These datasets can be accessed at the following sources:
https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset

REFERENCES

1. K. Alnowaiser, "Improving healthcare prediction of diabetic patients using KNN imputed features and tri-ensemble model," *IEEE Access*, vol. 12, pp. 16783–16793, 2024.
2. N. F. Cleymans, M. Van De Casteele, J. Vandewalle, A. K. Desouter, F. K. Gorus, and K. Barbé, "Analyzing Random Forest's predictive capability for type 1 diabetes progression," *IEEE Open Journal of Instrumentation and Measurement*, vol. 4, pp. 1–11, 2025.
3. J. Musa and A. M. Abdulazeez, "A review on diabetes classification based on machine learning algorithms," *The Indonesian Journal of Computer Science*, vol. 13, pp. –, Apr. 2024.
4. S. Salih and A. M. Abdulazeez, "Classification of diabetic retinopathy images through deep learning models – color channel approach: A review," *Indonesian Journal of Computer Science*, vol. 13, Feb. 2024.
5. L. Jia, Z. Wang, S. Lv, and Z. Xu, "PE_DIM: An efficient probabilistic ensemble classification algorithm for diabetes handling class imbalance missing values," *IEEE Access*, vol. 10, pp. 107459–107476, 2022.
6. L. Chaves and G. Marques, "Data mining techniques for early diagnosis of diabetes: A comparative study," *Applied Sciences*, vol. 11, p. 2218, Mar. 2021.
7. S. A. Antar, N. A. Ashour, M. Sharaky, M. Khattab, N. A. Ashour, R. T. Zaid, E. J. Roh, A. Elkamhawy, and A. A. Al-Karmalawy, "Diabetes mellitus: Classification, mediators, and complications; a gate to identify potential targets for the development of new effective treatments," *Biomedicine & Pharmacotherapy*, vol. 168, p. 115734, 2023.
8. M. S. Alam, M. J. Ferdous, and N. S. Neera, "Enhancing diabetes prediction: An improved boosting algorithm for diabetes prediction," *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2024. Online. Available: www.ijacsa.thesai.org.
9. M. Sinsirimongkhon, S. Arwatchananukul, and P. Temdee, "Multi-class classification method with feature engineering for predicting hypertension with diabetes," *Journal of Mobile Multimedia*, Feb. 2023.
10. K. Saiti, M. Macaš, L. Lhotská, K. Štechová, and P. Pithová, "Ensemble methods in combination with compartment models for blood glucose level prediction in type 1 diabetes mellitus," *Computer Methods and Programs in Biomedicine*, vol. 196, p. 105628, Nov. 2020.
11. G. Obaido, B. Ogbuokiri, C. W. Chukwu, F. J. Osaye, O. F. Egbelowo, M. I. Uzochukwu, I. D. Mienye, K. Aruleba, M. Primus, and O. Achilonu, "An improved ensemble method for predicting hyperchloremia in adults with diabetic ketoacidosis," *IEEE Access*, vol. 12, p. 9536–9549, 2024.
12. V. K. Daliya and T. K. Ramesh, "A cloud-based optimized ensemble model for risk prediction of diabetic progression—an azure machine learning perspective," *IEEE Access*, vol. 13, p. 11560–11575, 2025.
13. A. Daza, C. F. Ponce Sánchez, G. Apaza-Perez, J. Pinto, and K. Zavaleta Ramos, "Stacking ensemble approach to diagnosing the disease of diabetes," *Informatics in Medicine Unlocked*, vol. 44, p. 101427, 2024.

14. F. A. Khan, K. Zeb, M. Al-Rakhami, A. Derhab, and S. A. C. Bukhari, “Detection and prediction of diabetes using data mining: A comprehensive review,” *IEEE Access*, vol. 9, p. 43711–43735, 2021.
15. S. M. Ganie and M. B. Malik, “An ensemble machine learning approach for predicting type-ii diabetes mellitus based on lifestyle indicators,” *Healthcare Analytics*, vol. 2, p. 100092, nov 2022.
16. C. Al-Atroshi and A. M. Abdulazeez, “Predictions of early hospitalization of diabetes patients based on deep learning: A review: Machine learning,” *Indonesian Journal of Computer Science*, vol. 13, Feb. 2024.
17. Y. Li and W. Chen, “A comparative performance assessment of ensemble learning for credit scoring,” *Mathematics*, vol. 8, p. 1756, Oct. 2020.
18. I. D. Mienye and Y. Sun, “A survey of ensemble learning: Concepts, algorithms, applications, and prospects,” *IEEE Access*, vol. 10, p. 99129–99149, 2022.
19. I. Shaheen, N. Javaid, N. Alrajeh, Y. Asim, and S. Aslam, “Hi-le and hitcle: Ensemble learning approaches for early diabetes detection using deep learning and explainable artificial intelligence,” *IEEE Access*, vol. 12, p. 66516–66538, 2024.
20. M. S. Reza, R. Amin, R. Yasmin, W. Kulsum, and S. Ruhi, “Improving diabetes disease patients classification using stacking ensemble method with pima and local healthcare data,” *Heliyon*, vol. 10, p. e24536, Jan. 2024.
21. K. Oliullah, M. H. Rasel, M. M. Islam, M. R. Islam, M. A. H. Wadud, and M. Whaiduzzaman, “A stacked ensemble machine learning approach for the prediction of diabetes,” *Journal of Diabetes &amp; Metabolic Disorders*, vol. 23, p. 603–617, Nov. 2023.
22. M. Mohammed Rashid, O. M. Yaseen, R. Riyadh Saeed, and M. T. Alasaady, “An improved ensemble machine learning approach for diabetes diagnosis,” *Pertanika Journal of Science and Technology*, vol. 32, p. 1335–1350, Apr. 2024.
23. A. GR, A. Mary X, and M. Raja J, “Classification of diabetes using ensemble machine learning techniques,” *Scalable Computing: Practice and Experience*, vol. 25, p. 3172–3180, June 2024.
24. H. B. Kibria, M. Nahiduzzaman, M. O. F. Goni, M. Ahsan, and J. Haider, “An ensemble approach for the prediction of diabetes mellitus using a soft voting classifier with an explainable ai,” *Sensors*, vol. 22, p. 7268, Sept. 2022.
25. Z. Mushtaq, M. F. Ramzan, S. Ali, S. Baseer, A. Samad, and M. Husnain, “Voting classification-based diabetes mellitus prediction using hypertuned machine-learning techniques,” *Mobile Information Systems*, vol. 2022, p. 1–16, mar 2022.
26. F. Mazhar, W. Akbar, M. Sajid, N. Aslam, M. Imran, and H. Ahmad, “Boosting early diabetes detection: An ensemble learning approach with xgboost and lightgbm,” *Journal of Computing & Biomedical Informatics*, vol. 6, pp. 127–138, mar 2024.
27. M. Kawarkhe and P. Kaur, “Prediction of diabetes using diverse ensemble learning classifiers,” *Procedia Computer Science*, vol. 235, p. 403–413, 2024.
28. A. S. Aziz, K. Ibrahim, A. Elsharkawy, and N. Khaliel, “Anticipating diabetes using fusion-ensemble machine learning techniques,” *SciNexuses*, vol. 1, p. 44–51, May 2024.
29. H. Shao, X. Liu, D. Zong, and Q. Song, “Optimization of diabetes prediction methods based on combinatorial balancing algorithm,” *Nutrition &amp; Diabetes*, vol. 14, Aug. 2024.
30. J. Kaliappan, I. J. Saravana Kumar, S. Sundaravelan, T. Anesh, R. R. Rithik, Y. Singh, D. V. Vera-Garcia, Y. Himeur, W. Mansoor, S. Atalla, and K. Srinivasan, “Analyzing classification and feature selection strategies for diabetes prediction across diverse diabetes datasets,” *Frontiers in Artificial Intelligence*, vol. 7, Aug. 2024.