

Original Article

Enhancing Prediction Accuracy for Cardiovascular Heart Diseases through Optimized Ensembled Classifiers

Pratikkumar Parmar^{1,2}, Nirali Shah³, Vishal P. Jariwala⁴

^{1,3}Department of Electronics and Communication, Monark University, Gujarat, India.

^{2,4}Department of Electronics and Communication, Government Polytechnic Gandhinagar, Gujarat, India.

¹Corresponding Author : pratik.parmar24192@gmail.com

Received: 12 December 2025

Revised: 15 January 2026

Accepted: 19 February 2026

Published: 23 March 2026

Abstract - Cardiovascular Diseases (CVD) rank among the leading causes of global mortality. However, detecting CVD early and accurately can considerably reduce the rates of fatality. In order to support early diagnosis and intervention in cardiovascular health, Machine Learning (ML) models have arisen as powerful tools. A robust predictive system for cardiovascular heart disease using a suite of ML-based ensembled algorithms is presented in this study. A comprehensive heart disease dataset from IEEE DataPort is used to train the prediction models. The dataset is preprocessed by different processes, including data cleaning, standardization, and data balancing using the Synthetic Minority Oversampling Technique (SMOTE). This preprocessed dataset is bisected into a training dataset and a testing dataset. The ensembled machine learning algorithms, including AdaBoost (AdB), Gradient Boosting (GB), XGBoost (XGB), Extra Tree Classifier (ET), and Random Forest (RF), are trained using the training dataset for the prediction process. Hyperparameters of these algorithms are tuned using RandomSearchCV with 5-fold cross-validation to optimize the classifier algorithms and improve the efficiency of the prediction process. Standard performance parameters, accuracy, recall, precision, F1-score, and ROC-AUC, are used for the evaluation of the performance of each optimized model and compared with state-of-the-art model performances without optimization. Among all models, the optimized RF classifier demonstrated superior performance, achieving the highest accuracy of 96.31%, outperforming all other ensembled classifiers.

Keywords - CVD, Ensembled, Bagging, Boosting, SMOTE, RandomSearchCV.

1. Introduction

The heart is a key organ within the cardiovascular system, which facilitates the blood transport through an extensive network of arteries, veins, and capillaries. Heart disease, particularly Cardiovascular Disease (CVD), is the foremost cause of morbidity and mortality worldwide, claiming approximately 17.9 million lives annually [1-3]. This staggering figure represents nearly 31% to 32% of all global deaths, posing a significant threat to public health safety and the stability of national economies [4, 5].

Factors such as sedentary lifestyles, tobacco smoking, poor dietary habits, and genetic predisposition contribute to the emergence of chronic conditions like hypertension, diabetes mellitus, and high cholesterol, which further elevate cardiovascular risk [6]. These lifestyle-related disorders often lead to serious complications such as heart failure, stroke, cardiac arrest, and coronary artery disease. Socio-economic conditions play a pivotal role, as research indicates that 80% of fatalities occur in low- and middle-income countries where limited access to healthcare resources and medical personnel hinders effective disease management [7].

The requirement for early and accurate detection of cardiovascular symptoms is critical, as approximately 80% of CVD cases are preventable if identified at an incipient stage [8]. Traditional diagnostic procedures, including physical examinations, angiography, MRI, and transthoracic echocardiography, are frequently costly, time-consuming, and require highly specialized expertise, making them less accessible to underserved communities [9]. Furthermore, manual clinical decisions relying on specialist intuition are prone to undesirable biases, human error, and misinterpretation of test results, which can lead to fatal consequences [10]. Consequently, there is an urgent demand for reliable, cost-effective, and non-invasive automated medical diagnosis systems that can provide rapid results to support healthcare providers in making informed decisions.

The generalized solution proposed across the sources focuses on leveraging Artificial Intelligence (AI), specifically Machine Learning (ML) and Deep Learning (DL), to extract valuable insights and hidden patterns from vast medical databases [2, 9, 11]. Ensemble learning techniques-including bagging, boosting, stacking, and majority voting-emerge as



the most robust strategy for enhancing prediction performance. These methods combine the predictions of multiple diverse classifiers to maximize accuracy and minimize individual model blind spots [3, 12] Hybrid models integrating CNN, LSTM, and GRU are also frequently employed to capture deeper feature relationships [7, 13].

Moreover, the effectiveness of these predictive systems is significantly enhanced by rigorous data preprocessing [6, 14, 15]. Additionally, the application of hyperparameter optimization using tools like GridSearchCV, RandomSearchCV, or Optuna ensures that the models achieve their optimal performance benchmarks [16]. These integrated methodologies provide high-performance diagnostic frameworks, successfully transforming clinical data into reliable prognostic knowledge.

2. Related Work

Many researchers have worked on the prior prediction of heart diseases based on publicly available datasets and developed various methods of prediction.

[3] The research is based on the dataset of the UCI Heart Disease, with 14 original features. The authors came up with a new concatenated hybrid ensemble voting classifier, which combines two independent sub-ensembles with SVM, DT, KNN, LR, Adaboost, and NB. The final proposed model was carried out using GridSearchCV with five-fold cross-validation to find the highest efficacy, having the maximum accuracy of 86.89%, F1 Score of 84.3%, and sensitivity of 87.1%.

[4] A clinical dataset of heart diseases used in the study is found in the UCI Machine Learning repository and consists of 1329 instances and 14 independent variables. Having preprocessed the data. Gradient Boosting became the highest performer with an accuracy rate of 92.20%.

[7] used UCI machine learning library and boosting ensemble boosters, including AdaBoost, CatBoost, Gradient Boosting, Light GBM, XGBoost boosters, and Random Forest and Decision Trees. Addressing the overfitting problem, final outcomes suggest AdaBoost as the most promising model, whose test accuracy is significant (95%).

The research in [9] employed SVM, RF, LR, DT, KNN, and Adaboost with hyperparameter optimization through the Grid and Random search methods. The highest score was obtained by the SVM model, which scored 92 percent accuracy and an AUC of 0.91, which has been confirmed on various datasets.

The data used in the [10] study is a combination of heart disease data, which includes the records of 1190 patients. An innovative ensemble voting model has been produced through

an integration of the most prominent three classifiers, RF, XGBoost, and GB Machine, with the hyperparameter being optimized through the application of GridSearchCV to produce an accuracy of 96.17%.

[11] used dimensionality reduction that applied Principal Component Analysis (PCA) and Particle Swarm Optimization (PSO), Deep Learning with parameter optimization demonstrated the best accuracy of 84.47%.

The experiment in [12] used AdaBoost, XGBoost, RF, and Extremely Randomized Trees and hyperparameter optimization search based on RandomizedSearchCV and 5-fold cross-validation. The end results are AdaBoost as the best, with a testing accuracy of 93.70% and an F1 score of 93.69%.

The two datasets used in [15] are the IEEE Dataport and the UCI Kaggle Cleveland. The maximum prediction accuracy of 97% on all datasets with the six machine learning algorithms, such as KNN and RF, was achieved when combined through a soft-voting ensemble.

[17] The research dataset used is a 70,000-sample Cardiovascular Disease dataset, which is on Kaggle and has 13 clinical attributes. Data preprocessing was done, including missing values and outlier removal, Standard Scaler normalization, calculation of BMI, and Pearson correlation analysis to discard redundant features. Authors compared the methods of classification, RF, KNN, DT, XGBoost, and deep learning models, DNN, and KDNN. Each model was tuned on hyperparameters. The weighted majority voting with the model of ML Ensemble gave the highest score of 88.70% accuracy.

[18] applied an ensemble stacking technique on KNN, Naive Bayes (NB), Linear Discriminant Analysis (LDA), and DT as the base classifiers. To reduce the chances of misclassification, the model was advanced by feeding the outputs of the base classifier into SVM and RF meta-classifiers. The last outcomes show that diabetes and cardiovascular disease are more accurately represented by a value of 97.35% and 88.71%, respectively, than all the standalone individual models.

[19] The research relies on Framingham cardiovascular data consisting of 4,238 patients who employed SMOTE to balance the classes. The Hybrid Soft Voting Model using a two-step hybrid ensemble process was the best, with a maximum accuracy of 95.87%.

[20] This study involves a dataset of 16,068 samples from a Thai hospital, and classification was carried out using a Gradient Boosting (GB) based model and compared with DT, SVM, KNN, RF, and Extra Trees, where the GB model had a

multiclass classification accuracy of 83.81% when advanced feature engineering was applied.

[21] The analysis makes use of the Cleveland Clinic Heart Disease Dataset on Kaggle that consists of six most relevant clinical characteristics. LR, DT, and SVM techniques of classifying the heart disease prognosis were the main classification methods analyzed, with the LR providing the highest accuracy of 88.52%.

[22] The research makes use of Data preprocessing in order to give high-quality, secure input. The researchers used (KNN) and (SVM) as the main classification methods of CVD prediction that demonstrated a significant improvement in performance, with KNN reaching 95.38% and SVM reaching 90% after GA optimization.

This study shows that researchers have implemented a variety of ML algorithms for CVD prediction. However, state-of-the-art ML algorithms are struggling with the accuracy of prediction, which can be enhanced by supporting processes. Therefore, different data preprocessing techniques can be implemented to improve the quality of the dataset. Besides this, ML algorithms can be optimized to improve the efficiency of the prediction process.

Motivated by the above facts, this work incorporates a novel prediction framework that includes data preprocessing techniques, such as data cleaning, data balancing, and data scaling, to make the dataset ready to train the ML algorithms for the prediction process. This process also optimized ML algorithms by tuning the hyperparameters of the algorithms to enhance the efficiency of the prediction process. Results are compared before and after applying preprocessing and hyperparameter optimization techniques, and with similar research work.

3. Materials and Methods

This study follows a well-structured process to analyze data related to heart disease, as shown in pseudo-code below. It begins with using a combined dataset of five well-known datasets based on 11 shared features from IEEE Dataport (D), including 1190 samples. Then the workflow shifts towards preprocessing of the dataset, including cleaning the dataset to remove any errors or inconsistencies.

To make the data suitable for ML models, balancing and scaling techniques are applied, ensuring it remains balanced and consistent. The preprocessed dataset (Dpp) is divided into training (Dpptrain) and testing (Dpptest) datasets. Using this dataset, the training and testing process is carried out for several ensembled machine learning algorithms (C) to evaluate their ability to predict heart disease accurately. Hyperparameters of algorithms are fine-tuned to enhance the performance of each model. Key Performance Parameters (PP), including accuracy, recall, precision, F1-score, and

ROC-AUC are used to compare results, as shown in the Pseudo Code. Ultimately, the objective is to identify the most effective and reliable ML method to predict heart disease, based on how well each performs across these parameters.

Pseudo Code:

```

1: Dataset D
2: D -> Dpp (Data cleaning, balancing, scaling)
3: Dpp -> Dpptrain, Dppptest (80:20) splitting
4: for C in (Adb, GB, XGB, RF, ET)
    RandomSearchCV(Ci(Dpptrain),
                    5-fold CV, i <- niter)
    Cn (with best training Accuracy)
    PPC <- Cn(Dppptest)
5: Find the best C based on PP values

```

3.1. Dataset

In this work, the Comprehensive Heart Disease Dataset from IEEE Dataport is used [23]. It is a comprehensive mixture of independently available five popular heart disease datasets, namely Cleveland, Switzerland, Statlog, Long Beach VA, and the Hungarian (Heart) dataset. This dataset consists of 11 independent features, each having 1190 samples, including sex, age, cholesterol, type of chest pain, Maximum heart rate, resting BP, resting ECG, Fasting BS, old peak, exercise angina, ST slope, and 1 target feature [24]. Out of these 11 features, 6 are categorical features, and 5 are continuous-valued features. It includes a total of 561 records having heart disease, which are marked as 1 in the target class.

3.2. Data Pre-Processing

Data pre-processing is required to make the raw data ready to be used by an ML model. Raw data often contains noise, missing values, or irrelevant features that can negatively impact model accuracy. This process improves the quality of the dataset, enhances model performance, and reduces computational cost. Various steps of data preprocessing are as follows:

3.2.1. Data Cleaning

In the data cleansing process, samples with erroneous data are identified, which have missing data, outliers, and other incorrect data values [25]. This dataset does not have any samples with missing values. Cholesterol, RestingBPS, and Old peak features have outliers. Only the farthest outliers are removed, but soft outliers are considered part of the dataset because those datapoints contribute to increasing the model's generalization. In the Cholesterol feature, many samples have a 0 value, which is not practically possible; besides this, the 0 value of restbps and the negative value of oldpeak are not possible. So those samples are removed from the dataset, leaving 1018 samples in the dataset.

3.2.2. Data Balancing

Class misbalancing often biases toward the dominant class while neglecting minority cases. Such an imbalance may

result in degradation of prediction accuracy, especially in critical applications such as medical diagnosis or fraud detection, where minority classes often play a vital role. To overcome this challenge, the Synthetic Minority Over-sampling Technique (SMOTE) was introduced as an effective data-level approach [26]. SMOTE improves the presence of the minority class by creating synthetic artificial data points. SMOTE generates minority samples by finding K nearest neighbours, out of which a required number of samples are selected at random to balance the dataset [27, 28]. In this work, the SMOTE technique is used to balance the dataset. After using this technique, the dataset has an equal number of instances, with a total of 1082 samples, having 541 of each heart disease positive and negative class.

3.2.3. Data Scaling

Data scaling is a process to bring feature values into an equal range. Various techniques like standardization, Normalization, and min-max scaling can be used to scale the data [29]. In this work, the Standard Scaler standardization technique is used to scale the data as per Equation (1), where μ is the mean of the feature and σ is the standard deviation of feature. It brings the feature values on the same level, having a mean of 0 and a standard deviation of 1, allowing all features to contribute equally in the process.

$$x_{standard_scale} = \frac{x_i - \mu}{\sigma} \quad (1)$$

3.2.4. Data Splitting

The full dataset is not directly utilized to train the model. It is randomly divided into two parts, out of which one subset trains the model, while the other evaluates its performance. The dataset can be divided into two categories with different training-to-testing ratios. In this work, an 80:20 training-to-testing ratio is used, based on which 80% of the total dataset serves as training data (Dpptestrain), while the remaining 20% dataset (Dpptest) serves as testing data. So, considering the raw dataset with 1190 samples, it will be split into 952 samples for training and 238 samples for testing. After preprocessing, the dataset was reduced to 1082 samples, comprising 865 samples for training and 217 samples for testing, based on an 80:20 splitting ratio.

3.3. Ensembled Machine Learning Techniques

Ensemble learning is a machine learning method where a number of models, also known as base learners or weak learners, are trained to answer the same problem and then combined to give a better predictive performance on a problem than the individual models do. In this work, the classification process is performed by the Boosting and Bagging algorithms implementation.

Boosting Algorithms:

Boosting is a technique in the strategy of ensemble learning that tries to create a robust classifier by consolidating various weak classifiers in a bid to reduce training mistakes.

Boosting models are trained one after the other, and each new model concentrates on the errors that the previous model committed. This learning process is an iterative process, and through this process, the ensemble is able to enhance its predictive performance. In this paper, Adaptive Boosting (AdB), Gradient Boosting (GB), and Extreme Gradient Boosting (XGB) boosting algorithms are used to increase the classification performance [8].

AdaBoost: AdaBoost is a machine learning algorithm that uses a boosting process to generate a strong classifier by combining multiple weak learners, considering weighted linear aggregation. The algorithm uses an iterative learning approach whereby the training elements are re-weighted in succession. The first one is that each case is given equal weight; in every other iteration, the weight of misclassified cases is increased, and the weight of correctly classified cases is reduced. The procedure causes the consecutive base classifiers to concentrate on challenging samples. The last classification model is considered to be a weighted average of all the classifiers produced throughout the iterations. Even though AdaBoost is effective in enhancing the performance of the classifier, the algorithm is vulnerable to outliers and noisy data because it focuses on the erroneous samples [4, 6].

Gradient Boosting: Gradient boosting is based on the underlying principle of adding more models to the already existing models to address the mistakes of the earlier models. The method is based on calculus-based optimization, or computing the gradient and partial derivatives of a specified loss function. The choice of the loss function defines the appropriate gradient expressions, and a popular loss function is the cross-entropy loss when dealing with a classification problem. Even though the mathematical formulations may be complex, the end goal is similar: to continuously reduce the loss function by updating the model towards the negative gradient direction, and this way continuously increasing predictive accuracy [7, 10].

XGBoost is a new state-of-the-art gradient-boosted decision tree model that is aimed at improving the performance of weak learners. It uses a greedy algorithm to arrive at the best split points in a tree-building process. First, a poor classifier is trained on the data, and then additional classifiers are added one at a time to enhance the overall model performance without deleting the input of other learners. XGBoost applies gradient descent optimization in order to minimize the loss function when adding new models, so that each new classifier works on addressing the errors committed by the previous classifier. With this process of iteration, the algorithm will gradually decrease the distance between the actual prediction and the actual loss by moving in the direction of the negative gradient of the loss function. It brings in regularization methods used to regularize the loss and avoid overfitting, leading to better control of the model complexity [2, 7].

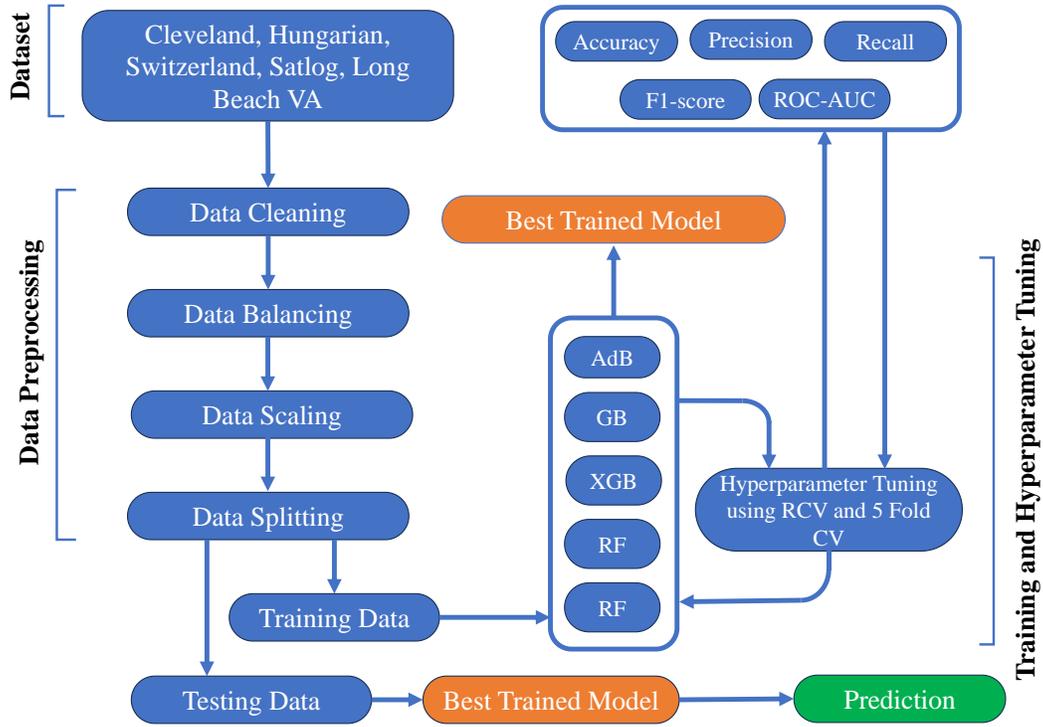


Fig. 1 Flow of the prediction process

Bagging Algorithms: This method involves training several copies of the same base model in parallel, using different bootstrap samples and combining the results later in an aggregation process to obtain the final result [8].

Random Forest: This is a type of classifier that is based on an ensemble learning family of bagging, and it involves the use of multiple decision tree models to improve the prediction. It builds many decision trees based on bootstrap samples on the training data. Entropy and Gini impurity of any attribute are computed as per the following Equations (2) and (3).

$$\text{Gini Index} = 1 - \sum_{i=1} P_i^2 \quad (2)$$

$$\text{Entropy} = - \sum_{i=1} P_i \log_2(P_i) \quad (3)$$

The attribute that offers less uncertainty is chosen to split at the node [12, 15].

Extra Tree: Another tree-based ensemble algorithm, Extremely Randomized Trees (ET), is also an algorithm that can be used instead of Random Forests in two aspects. To begin with, ET solely chooses split thresholds randomly and does not even seek out the best places to split the node. Second, it does not apply bootstrap sampling to build trees as in the case of Random Forests, but it employs the whole training dataset. The rationale behind the ET algorithm is that the randomization of split selection can be used to further

reduce model variance. In addition, there is no bootstrap sampling, which can make the bias less, and hence there is a better generalization performance. [2,20]

3.4. Hyperparameter Tuning

All forms of classifiers cannot do without hyperparameter tuning, which is essential in improving model performance and making more accurate and reliable predictions. [10] The best training hyperparameters of each of the classifiers are identified by employing the hyperparameters of the RandomSearchCV to obtain a better predictive performance in the proposed system. RandomSearchCV systematically fits the machine learning models with a set of combinations of values of the hyperparameters chosen within the available search space and determines the optimal set of hyperparameter values by the chosen evaluation measure. [30] Although it does not verify the optimum solution, it is computationally rapid relative to other optimizations, such as GridSearchCV, which examines all the potential combinations in the search space and verifies the best optimum result, but is computationally expensive and time-consuming. [2] RandomSearchCV nearly validates the optimal or close optimum solution without a large amount of computation and reduced time.

Both Grid Search and Randomized Search Cross-Validation make use of k-fold cross-validation to assess the level of model performance with each configuration of

hyperparameters. This method divides the data set into k equal folds, with each fold utilized once as a validation set and the rest of the folds being used as a training set [31]. k-fold cross-validation estimates the model performance with a high degree of accuracy by averaging the performance of all folds, and reduces the likelihood of overfitting. It is also possible to have more confidence in the ability of the model to generalize.

Table 1. Result of ML models with raw data (in %)

	Accuracy	Precision	Recall	F1 Score	ROC Score
AdB	84.03	85.27	85.27	85.27	83.92
GB	89.92	92.00	89.15	90.55	89.18
XGB	92.44	93.70	92.25	92.97	92.12
RF	93.70	93.18	95.35	94.25	93.70
ETC	94.12	94.57	94.57	94.57	94.08

3.5. Evaluation Parameters

All ML algorithms mentioned in the previous section provide the result in various parameters like Accuracy, Sensitivity, Specificity, and F-1 Score. All these parameters can be derived from the values of the confusion matrix as per the following Equations (4), (5), (6), and (7) [32]:

$$Accuracy = \frac{TPs + TNg}{TPs + FPs + FNg + TNg} \quad (4)$$

$$Precision = \frac{TPs}{TPs + FPs} \quad (5)$$

$$Recall = \frac{TPs}{TPs + FNg} \quad (6)$$

$$F1-Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (7)$$

Where classified instances are divided into four categories, namely True Positive (TPs), True Negative (TNg), False Positive (FPs), and False Negative (FNg).

3.6. Experimental Setup

For this work, the Python platform is used for evaluating different algorithms. Python is a strong programming language known for its flexibility and rich library support, like Scikit-learn, NumPy, and Pandas, which support a wide range of ML algorithms. The flow of the prediction process is shown in Figure 1.

4. Results and Analysis

4.1. Performance Parameters

This work has used all the features of the Comprehensive Heart Disease dataset from IEEE Dataport. Results are recorded in two conditions, one with raw data and another with processed data and hyperparameter tuning. Six state-of-the-art ML algorithms are trained and tested using this dataset. Classified instances are divided into TPs, FPs, FNg, and TNg.

AdB, GB, XGB, RF, and ET have classified 200, 214, 220, 223, and 224 samples correctly, including TPs and TNg, out of 238 testing samples, which are 20% of the total raw data with 1190 samples. ET has correctly classified the highest 224 samples, whereas AdB has classified only 200 samples correctly, which is the lowest among all other algorithms. RF and XGB performed almost equally, classifying 223 and 220 samples correctly, respectively, while GB classified 214 correctly. ET leads all other algorithms with a classification accuracy of 94.12%, leaving behind RF with 93.70%. GB and XGB obtained 89.92% and 92.44% accuracy for the test set, respectively. AdB legs with an accuracy value of 84.03%. All accuracies mentioned above are calculated for the testing set. Performance comparison of all ML models is shown in Table 1.

		Actual Class	
		Positive	Negative
Predicted Class	Positive	99	16
	Negative	14	88

(a)

		Actual Class	
		Positive	Negative
Predicted Class	Positive	111	9
	Negative	2	95

(b)

		Actual Class	
		Positive	Negative
Predicted Class	Positive	111	7
	Negative	2	97

(c)

		Actual Class	
		Positive	Negative
Predicted Class	Positive	111	6
	Negative	2	98

(d)

		Actual Class	
		Positive	Negative
Predicted Class	Positive	110	7
	Negative	3	97

(e)

Fig. 2. Confusion matrix of classification by different classifiers after optimization and data preprocessing

After data cleaning by removal of extreme outliers and data balancing by the STOME technique, the data is scaled by the standard scaler method, with a total of 1082 samples. Then, all classifier algorithms are optimized using RandomSearchCV by tuning the hyperparameters of the classifier with 5-fold cross-validation. 'n_iter' is set to 100 for RandomSearchCV. Classifiers examined in this study are mainly tree-based classifiers. Tuned hyperparameter values by RandomSearchCV for AdaBoost are 'n_estimators'=56 and 'random_state'=23, for GB 'n_estimators' = 93, 'max_depth'= 5 and 'random_state' = 65, for XGB 'n_estimators' = 96, 'max_depth'= 4 and 'random_state' = 97.

For RF 'n_estimators' = 92, 'max_depth'= 14 and 'random_state' = 77 and for ET 'n_estimators' = 37, 'max_depth' 14 and 'random_state' = 88 are set for the best result.

Table 2. Result of ML models with optimization (in %)

	Accuracy	Precision	Recall	F1 Score	ROC Score
AdB	86.18	86.09	87.61	86.84	86.11
GB	94.93	92.50	98.23	95.28	94.79
XGB	95.85	94.07	98.23	96.10	95.75
RF	96.31	94.87	98.23	96.52	96.27
ETC	95.39	94.02	97.35	95.65	95.31

After the hyperparameter tuning, the categorization of predicted samples is shown in the form of a confusion matrix in Figure 2. It shows that the correctly classified samples increase in number for all classifiers. RF with optimization has shown the highest number of truly classified samples among all other algorithms. RF has correctly classified 209 out of 217 samples with an accuracy of 96.31 %, precision 94.87% and recall 98.23%, which is the highest among all other algorithms. XGB, ET, and GB follow the RF closely, achieving accuracy of 95.85%, 95.39% and 94.93% respectively.

XGB and ET show almost similar precision value of 94.07% and 94.02% respectively, while GB, XGB, and RF show the same recall value of 98.23% but ET lags behind with 97.35%. RF also performs well in F1-score with a value of 96.52% slightly leaving behind XGB, ET, and GB with value 96.10%, 95.65% and 95.28% respectively. Performance of AdaBoost is relatively lower in all parameters, with the lowest accuracy of 86.18% and a recall of 87.61%. Performance parameters, including accuracy, recall, precision, F1-Score, and ROC-AUC of all ensemble algorithms, are shown in Table 2.

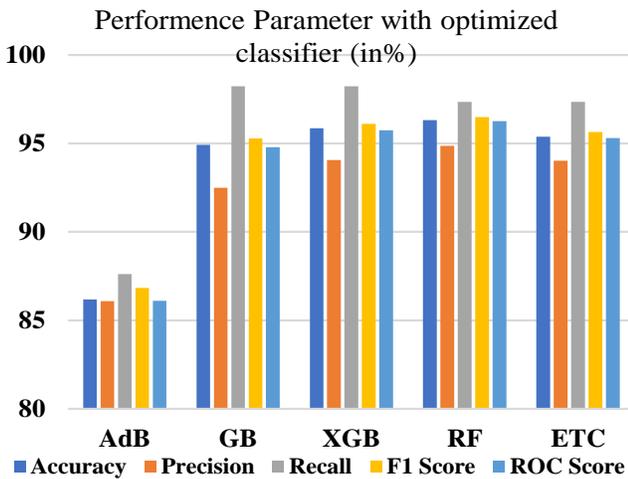


Fig. 3 Comparison analysis of models with preprocessed data and optimized algorithms

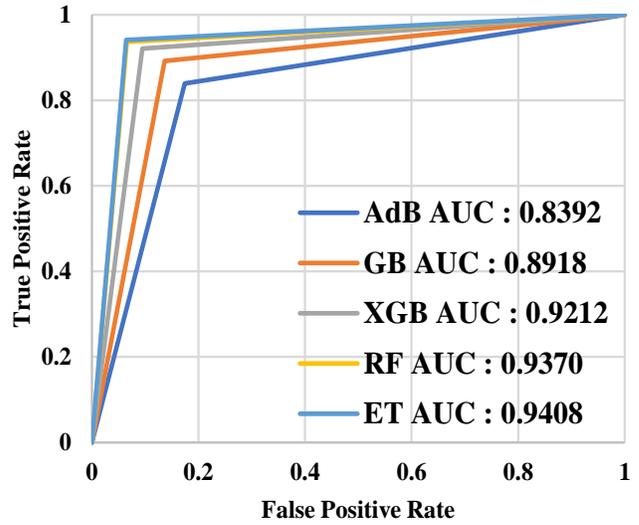


Fig. 4 ROC-AUC with raw data and default algorithms

Comparison of all ML classifiers is shown in Figure 3. It can be seen that RF not only correctly classifies maximum instances but also maintains a strong balance between false positives and false negatives, making it highly trustworthy.

4.2. ROC-AUC

The Receiver Operating Characteristic (ROC) curve illustrates how well a classifier performs with varying threshold values. It shows the relation between true positive rate and true negative rate, whereas the Area Under the Curve (AUC) summarizes the ROC in a scalar value. A higher AUC value indicates a higher classifier capacity to differentiate between two classes [30]. Here, ROC curves and their respective AUC values of ML classifiers with raw data and pre-processed data with optimized classifiers are shown in Figures 4 and 5, respectively.

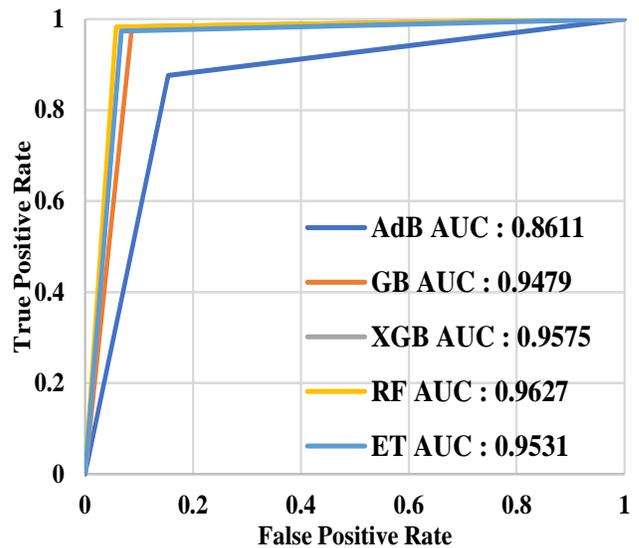


Fig. 5 ROC-AUC with preprocessed data and optimized algorithms

Table 3. Comparison of results with peers

Ref.	Year	Classification Model	Dataset	Performance
[12]	2025	AdB, XGB, RF, ET	Kaggle comprehensive with 1190 samples	AdB average testing Accuracy of 93.70%
[22]	2025	KNN, SVM	Kaggle comprehensive with 1192 samples	95.38% with GA optimized KNN
[7]	2024	Decision Tree, RF, CatBoost, XGB, AdB, Light GBM	Kaggle with 8763 and 26 attributes	With AdaBoost, the Accuracy is 95.2%.
[9]	2024	DT, RF, SVM, LR, AdB, KNN	Cleveland and Statlog heart disease datasets	SVM, with the Cleveland dataset: 92%, with Satlog Dataset: 91.31% Accuracy
[3]	2023	E1: SVM, DT, KNN E2: LR, AdB, NB then concatenated ensemble	UCI Kaggle dataset with 920 samples	Accuracy of 86.89%
[24]	2023	KNN, LR, NB, RF, GB, AdB	IEEE Dataport Heart disease dataset (comprehensive)	With the soft voting ensembled method, 95.00% Accuracy
[33]	2022	RF, ET, GB, XGB	IEEE Dataport Cardiovascular Disease (CVD) dataset	With a stacking 92.34% Accuracy
Proposed work		AdB, GB, XGB, RF, ET	Heart Disease Dataset from IEEE Dataport with data cleaning, standardization, and SMOTE data balancing	The highest accuracy of 96.31% with an optimized RF classifier using RandomSearchCV with 5-fold CV

As per Figure 4, the AUC of ET exhibited a favourable AUC score of 0.9408, while RF and XGB obtained an AUC score of 0.9370 and 0.9212. GB achieved an AUC score of 0.8918, showing good performance. AdB achieved an AUC of 0.8392. The effect of data pre-processing and optimization can be observed after tuning the hyperparameters as per Figure 5. The AUC of the AdB improved slightly to 0.8611. GB achieved a 0.9479 AUC score. The XGB showed improvement by reaching an AUC score of 0.9575, surpassing the ET score of 0.9531, after tuning. Particularly, RF achieved a significant rise in AUC, reaching 0.9627. Results show that RF excels in the classification process of CVD prediction. Comparison of results with peers, having similar methodologies and algorithms, is shown in Table 3.

5. Conclusion

The cardiovascular heart disease prediction framework presented in this research showcases the effectiveness of ensembled ML algorithms in anticipating the onset of cardiovascular diseases. The predictive model is constructed based on six different ML classifiers. Before training, the dataset underwent preprocessing, including cleaning, scaling, balancing, and partitioning into training and testing subsets. Classifier training and evaluation are accompanied by

performance optimization through hyperparameter tuning using randomized search techniques. Among the five classifiers, the optimized RF algorithm delivered the most favorable results, attaining an accuracy score of 96.31%, a recall score of 98.23%, an F1-score of 96.52%, and a precision score of 94.87%. This shows that the pre-processed data produces better results compared to raw data, as the raw data can be erroneous. This work also focuses on the importance of optimization. A properly optimized algorithm with tuned hyperparameters can perform better in the classification process. Another important aspect addressed here is the reduction of False Negatives (FNg). Since this is a medical prediction process, misclassifying an actual positive case as negative can lead to serious real-world consequences. This type of misclassification falls under False Negatives (FNs). The results demonstrate that RF effectively minimizes false negatives, helping to mitigate potential real-life risks. This work highlights the potential use of ML-based systems in early prediction and risk stratification for cardiovascular diseases, based on which a low-cost software application can be developed for early detection of cardiovascular diseases. Future scope may involve leveraging larger and diverse datasets, integrating stacking models and deep learning algorithms for better performance.

References

- [1] Ankit Kumar, Kamred Udham Singh, and Manish Kumar, "A Clinical Data Analysis Based Diagnostic Systems for Heart Disease Prediction Using Ensemble Method," *Big Data Mining and Analytics*, vol. 6, no. 4, pp. 513-525, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]

- [2] Daniyal Asif et al., “Enhancing Heart Disease Prediction through Ensemble Learning Techniques with Hyperparameter Optimization,” *Algorithms*, vol. 16, no. 6, pp. 1-17, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [3] Annwasha Banerjee Majumder et al., “Heart Disease Prediction Using Concatenated Hybrid Ensemble Classifiers,” *Algorithms*, vol. 16, no. 12, pp. 1-19, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Shahid Mohammad Ganie et al., “An Improved Ensemble Learning Approach for Heart Disease Prediction Using Boosting Algorithms,” *Computer Systems Science and Engineering*, vol. 46, no. 3, pp. 3993-4006, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [5] Xiao-Yan Gao et al., “Improving the Accuracy for Analyzing Heart Diseases Prediction Based on the Ensemble Method,” *Complexity*, vol. 2021, no. 1, pp. 1-10, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Joshua Emakhu, Sujeet Shrestha, and Suzan Arslanturk, “Prediction System for Heart Disease Based on Ensemble Classifiers Prediction System for Heart Disease Based on Ensemble Classifiers,” *Proceedings of the 5th NA International Conference on Industrial Engineering and Operations Management Detroit*, Michigan, USA, pp. 2337-2347, 2020. [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Najmu Nissa, Sanjay Jamwal, and Mehdi Neshat, “A Technical Comparative Heart Disease Prediction Framework Using Boosting Ensemble Techniques,” *Computation*, vol. 12, no. 1, pp. 1-22, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Deepali Yewale, S.P. Vijayaragavan, and V.K. Bairagi, “An Effective Heart Disease Prediction Framework based on Ensemble Techniques in Machine Learning,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, pp. 182-190, 2023. [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Mohammed Amine Bouqentar et al., “Early Heart Disease Prediction using Feature Engineering and Machine Learning Algorithms,” *Heliyon*, vol. 10, no. 19, pp. 1-23, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Sohaib Asif et al., “Improving the Accuracy of Diagnosing and Predicting Coronary Heart Disease using Ensemble Method and Feature Selection Techniques,” *Cluster Computing*, vol. 27, pp. 1927-1946, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Agung Muliawan, Achmad Rizal, and Sugondo Hadiyoso, “Heart Disease Prediction based on Physiological Parameters Using Ensemble Classifier and Parameter Optimization,” *Journal of Applied Engineering and Technological Science*, vol. 5, no. 1, pp. 258-267, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [12] Enoch Sakyi-Yeboah et al., “Heart Disease Prediction Using Ensemble Tree Algorithms: A Supervised Learning Perspective,” *Applied Computational Intelligence and Soft Computing*, vol. 2025, no. 1, pp. 1-18, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] Ahmed Almulihi et al., “Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction,” *Diagnostics*, vol. 12, no. 12, pp. 1-17, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [14] Sarria E. A. Ashri, M. M. El-Gayar, and Eman M. El-Daydamony, “HDPF: Heart Disease Prediction Framework Based on Hybrid Classifiers and Genetic Algorithm,” *IEEE Access*, vol. 9, pp. 146797-146809, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Mohammed Ahmed, and Idress Husien, “Hybrid Machine Learning Approach for Accurate Heart Disease Prediction,” *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 728-737, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] Priyanka Gupta, and D.D. Seth, “Improving the Prediction of Heart Disease Using Ensemble Learning and Feature Selection,” *International Journal of Advances in Soft Computing and its Applications*, vol. 14, no. 2, pp. 37-48, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [17] Abdullah Alqahtani et al., “Cardiovascular Disease Detection using Ensemble Learning,” *Computational Intelligence and Neuroscience*, vol. 2022, no. 1, pp. 1-9, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Asfandyar Khan et al., “Cardiovascular and Diabetes Diseases Classification using Ensemble Stacking Classifiers with SVM as a Meta Classifier,” *Diagnostics*, vol. 12, no. 11, pp. 1-15, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Sushree Chinmayee Patra, B. Uma Maheswari, and Peeta Basa Pati, “Forecasting Coronary Heart Disease Risk with a 2-Step Hybrid Ensemble Learning Method and Forward Feature Selection Algorithm,” *IEEE Access*, vol. 11, pp. 136758-136769, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Khomkrit Yongcharoenchaiyasit et al., “Gradient Boosting Based Model for Elderly Heart Failure, Aortic Stenosis, and Dementia Classification,” *IEEE Access*, vol. 11, pp. 48677-48696, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [21] G. Manikandan et al., “Classification Models Combined with Boruta Feature Selection for Heart Disease Prediction,” *Informatics in Medicine Unlocked*, vol. 44, pp. 1-12, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [22] Sherko H. Murad et al., “Hybrid Genetic Algorithms-Driven Optimization of Machine Learning Models for Heart Disease Prediction,” *MethodsX*, vol. 15, pp. 1-10, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [23] Mana Saleh Al Reshan et al., “A Robust Heart Disease Prediction System using Hybrid Deep Neural Networks,” *IEEE Access*, vol. 11, pp. 121574-121591, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [24] Nadikatla Chandrasekhar, and Samineni Peddakrishna, “Enhancing Heart Disease Prediction Accuracy Through Machine Learning Techniques and Optimization,” *Processes*, vol. 11, no. 4, pp. 1-31, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [25] Kaushalya Dissanayake, and Gapar Md Johar, “Comparative Study on Heart Disease Prediction using Feature Selection Techniques on Classification Algorithms,” *Applied Computational Intelligence and Soft Computing*, vol. 2021, no. 1, pp. 1-17, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]

- [26] Norma Latif Fitriyani et al., “HDPM: An Effective Heart Disease Prediction Model for a Clinical Decision Support System,” *IEEE Access*, vol. 8, pp. 133034-133050, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [27] Aqsa Rahim et al., “An Integrated Machine Learning Framework for Effective Prediction of Cardiovascular Diseases,” *IEEE Access*, vol. 9, pp. 106575-106588, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [28] Ayesha Noor et al., “Heart Disease Prediction using Stacking Model with Balancing Techniques and Dimensionality Reduction,” *IEEE Access*, vol. 11, pp. 116026-116045, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [29] Jian Ping Li et al., “Heart Disease Identification Method using Machine Learning Classification in E-Healthcare,” *IEEE Access*, vol. 8, pp. 107562-107582, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [30] Parmonangan R. Togatorop et al., “Optimizing Random Forest using Genetic Algorithm for Heart Disease Classification,” *Lontar Komputer: Scientific Journal of Information Technology*, vol. 13, no. 1, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [31] Mirza Muntasir Nishat et al., “A Comprehensive Investigation of the Performances of Different Machine Learning Classifiers with SMOTE-ENN Oversampling Technique and Hyperparameter Optimization for Imbalanced Heart Failure Dataset,” *Scientific Programming*, vol. 2022, no. 1, pp. 1-17, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [32] Abbas Jafar, and Myungho Lee, “HypGB: High Accuracy GB Classifier for Predicting Heart Disease with HyperOpt HPO Framework and LASSO FS Method,” *IEEE Access*, vol. 11, pp. 138201-138214, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [33] Achyut Tiwari, Aryan Chugh, and Aman Sharma, “Ensemble Framework for Cardiovascular Disease Prediction,” *Computers in Biology and Medicine*, vol. 146, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]