Feature elimination and stacking framework for accurate heart disease detection in IoT healthcare systems using clinical data

Wang Jian[1], Jian Ping Li[2]*, Amin Ul Haq[2]*, Shakir Khan[3,4], Reemiah Muneer Alotaibi[4], Saad Abdullah Alajlan[4] and Md Belal Bin Heyat[5]

[1]School of Artificial Intelligence, Neijiang Normal University, Neijiang, Sichuan, China, [2]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Sichuan, Chengdu, China, [3]University Centre for Research and Development, Department of Computer Science and Engineering, Chandigarh University, Mohali, India, [4]College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia, [5]CenBRAIN Neurotech Center of Excellence, School of Engineering, Westlake University, Hangzhou, Zhejiang, China

Introduction: Heart disease remains a complex and critical health issue, necessitating accurate and timely detection methods.

Methods: In this research, we present an advanced machine learning system designed for efficient and precise diagnosis of cardiac disease. Our approach integrates the power of Random Forest and Ada Boost classifiers, along with incorporating data pre-processing techniques such as standard scaling and Recursive Feature Elimination (RFE) for feature selection. By leveraging the ensemble learning technique of stacking, we enhance the model's predictive performance by combining the strengths of multiple classifiers.

Results: The evaluation metrics results demonstrate the superior accuracy and obtained the higher performance in terms of accuracy, 99.25%. The effectiveness of our proposed system compared to baseline models.

Discussion: Furthermore, the utilization of this system within IoT-enabled healthcare systems shows promising potential for improving heart disease diagnosis and ultimately enhancing patient outcomes.

KEYWORDS

heart disease, machine learning, classification, stacking, healthcare

1 Introduction

Heart disease (HD) is a serious public health problem that has affected millions of individuals worldwide according to the World Health Organization (WHO) (1, 2). Shortness of breath, muscle weakness, and swelling feet are prominent signs of HD (3). The diagnosis of HD is significantly important for patient treatment and recovery in the Medical Internet of Things system (MIoT) (4). Experts and medical specialists in MIoT systems have presented many non-invasive approaches for classifying and diagnosing cardiac disease (5). Machine learning (ML) and deep learning (DL) models are widely utilized in the design of computer-aided diagnosis systems (CAD) for the detection of heart disease (6).

Different heart disease diagnosis methods have been presented utilizing ML learning approaches in the literature. Detrano et al. (7) created an HD classification system utilizing ML algorithms. The Cleveland heart disease (CHD) dataset was used with global evolutionary and feature selection methods. Their proposed method recorded an accuracy of 77%. Humar et al. (8) proposed an HD detection method using a Neural Network (NN) and Fuzzy logic (FL). The classification accuracy of the said model was 87.4%. Palaniappan et al. (9) proposed a diagnosis method for HD diagnosis. The system was developed using ML models including Navies Bays (NB), Decision Trees (DT), and Artificial Neural Network (ANN). NB attained 86.12% accuracy, ANN achieved 88.12% accuracy, and 80.4% accuracy gained by the DT algorithm. Olaniyi et al. (10) proposed a three-phase model using the ANN for HD detection in angina that obtained an accuracy of 88.89%.

For the diagnosis of HD, Samuel et al. (11) designed an integrated model based on an ANN and Fuzzy AHP. In terms of accuracy, 91.10% was gained by the technique. Liu et al. (12) suggested a high-definition model based on Relief and rough set techniques. Their proposed method attained an accuracy of 92.32%. Mohan et al. (13) proposed an HD detection method using mixed ML algorithms. He also proposed a new strategy for selecting key features from data for effective machine learning classifier training and testing. They achieved 88.07% accuracy. Haq et al. (14) Proposed a machine learning-based diagnosis technique for identifying HD. ML models were used to detect HD. To choose the features, feature selection algorithms were utilized. For feature selection, they designed the Fast-Conditional-Mutual-Information (FCMIM) feature selection method. The proposed model (FCMIM-SVM) obtained a high accuracy of 92.37%. Tiwari et al. (15) proposed an ensemble approach for predicting cardiovascular illness. The framework (SE) employs a stacked ensemble classifier with machine learning algorithms such as ExtraTrees Classifier, Random Forest, and XGBoost. They have used different evaluation metrics for the proposed model (SE) evaluation. The proposed method obtained 92.34% accuracy.

The presented literature on the existing HD diagnosis models is shown in Table 1 in order to reach the problem gap in existing models in a systematic way. All of the prior treatments used a variety of methodologies to detect HD in its initial stages. However, all existing algorithms have low accuracy and are computationally complex to diagnose HD. The prediction accuracy of the HD detection approach, as shown in Table 1, requires significant enhancement for efficient and accurate detection of HD. Thus, the key concerns with the preceding methodologies are low accuracy and long computation times, which may be attributed to the usage of irrelevant features in the dataset. To solve these difficulties, new ways of identifying HD in IoT healthcare systems are necessary. Improving forecast accuracy is a major challenge and study area. Thus, the primary goal of this research is to develop an accurate and efficient HD diagnosis system.

In this research study, we have proposed an ML-based computer-aided diagnosis (CAD) approach for detecting HD early in the Medical Internet of Things (IoT) system. The objective is to develop a robust and efficient system that can assist healthcare professionals in accurately identifying HD in patients. In the designing of the CAD system, we applied data pre-processing

TABLE 1 Proposed models summary.

Model	FS	Data set	Acc (%)	Ref
ML algorithms	–	CHD	77	(7)
MLP + SVM	–	CHD	80.41	(16)
Hybrid MLmodel (HRFLM)	–	CHD	88.07	(13)
ANN + Fuzzy Logic (ANN-FL)	–	PID and CHD	87.4	(8)
ANN ensemble-based diagnosis system	–	CHD	89.01	(17)
IHDPS	–	–	88.12	(9)
3-phase technique using ANN	–	SCH.	88.89	(10)
XGBoost		CDHD	87.28	(18)
ANN-FUZZY-AHP	–	CHD	91.1	(11)
CART		HDD	87	(19)
RRS-HD	RFRS feature selection	SCH	92.32	(12)
HISFP	Relief, mRMR, LASSO	CHD	89	(2)
SVM		Cleveland Clinic dataset	96	(20)
FCMIM-SVM	Relief, mRMR, LASSO, and LLBFS	CHD	92.37	(14)
SE		Hungarian, Cleveland, Long Beach VA, Switzerland, and Statlog	92.34	(15)

techniques such as standard scalar and the removal of null values from the data set. To select related features from the data set, we incorporated the Recursive Feature Elimination (RFE) algorithm. This helps to balance the data for proper training of the algorithm and enhance the algorithm's predictive capability. The machine learning classifiers Random Forest (RF) and Ada Boost (AB) were used for the classification of affected and healthy control subjects. These models were trained and evaluated using the entire data set and selected feature data set. To further improve the predictive results of these models, we incorporated a stacking approach to select the best meta-classifier between the Random Forest and Ada Boost. We defined a parameter grid for grid search for both algorithms. Furthermore, a hold-out validation mechanism was utilized, and data were split for training and testing in portions of 80 and 20%, respectively. The Cleveland Heart Database was used to validate the proposed model. Different performance assessment

metrics were computed for model evaluation. The experimental results unequivocally demonstrated that our proposed model outperformed the baseline models in terms of predictive accuracy. Furthermore, its ease of use and compatibility with IoT healthcare systems make it an appealing and practical choice for heart disease prediction.

The innovative points of this research study are listed below:

- A CAD approach based on ML is designed to detect cardiac disease in its early stages in the MIoT systems.
- To normalize the dataset, we incorporated data preprocessing such as stander scalar and RFE algorithm for irrelevant feature elimination. The Random Forest and Ada Boost were trained and tested on entire selected feature datasets to classify heart disease and healthy control subjects.
- To further improve classification performance, the ensemble learning technique stacking was used to select the best meta-classifier between Random Forest and Ada Boost. The meta-classifier RF was used for the final classification.
- The proposed model performance was compared with baseline models, and our approach outperformed them. Hence, it is recommended for use in diagnosing heart disease in MIoT systems.

The structure of the remaining sections includes data collection and model methodology (Section 2), experiments (Section 3), discussion (Section 4), and conclusion (Section 5).

2 Research design

2.1 Data sets

The Cleveland heart disease dataset (CHD) (https://www.kaggle.com/datasets/aavigan/cleveland-clinic-heart-disease-dataset) is being examined for testing purposes in this study. Furthermore, for cross-validation of the models, we incorporated the data set Heart Statlog Cleveland Hungary (SCH) (https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive).

2.2 Methodology

The proposed methodology is described in the following subsections:

1) Recursive Feature Elimination (RFE) algorithm for feature selection: feature selection is the process of selecting a subset of relevant features from a larger set of available features in a dataset. It is a critical step in machine learning and data analysis, as it helps improve model performance, reduce overfitting, and enhance interpretability. Feature selection also reduces the computation time of machine learning Algorithm 1. REF is a feature selection technique commonly used in machine learning to identify the most relevant features in a dataset. It aims to find the subset of features that are most relevant to a given machine learning task. It starts by taking a feature matrix X of shape (n samples, n features) and a target variable y of shape (n samples) as input. Additionally, a

machine learning model is chosen to perform the feature selection process.

The RFE algorithm begins by initializing an empty list called "selected features" to store the indices of the selected features. It also creates another list of remaining features, which initially contains all the indices of the features in the "original feature" matrix.

The algorithm enters a white loop that continues until the number of selected features in selected features reaches the desired target number of features N. Inside the loop, the model is trained using the trained model and gets importance scores procedure.

This procedure fits the model on the subset of features given by X [: remaining features] and y. It then calculates the importance scores for each feature using a specific method provided by the chosen model. The importance scores represent the relevance or contribution of each feature to the model's performance.

Next, the algorithm utilizes the least important feature procedure to identify the index of the least important feature based on the importance scores. This feature is then appended to the selected feature list and removed from the remaining feature list. The algorithm proceeds by selecting the subset of features from the original feature matrix X using the indices in the selected feature list, resulting in a new matrix called X selected. The model is then retrained using this reduced feature set by applying the train model procedure, which fits the model on selected X and y. The loop continues until the number of selected features reaches the target number N. At this point, the algorithm terminates, and the selected features list contains the indices of the optimal feature subset, according to the RFE algorithm. The RFE algorithm offers several advantages, including improved model interpretability, enhanced generalization capabilities, and reduced overfitting. By iteratively eliminating the least important features and retraining the model, RFE enables the identification of the most informative features for the given task, leading to more accurate and efficient models.

Pseudo-code for the Recursive Feature Elimination (RFE) algorithm is shown in Algorithm 1.

2.3 Proposed classification algorithms

2.3.1 Random Forest ensemble learning algorithm

Random Forest (RF) (21) is an ensemble learning algorithm that combines multiple decision trees to make predictions. It is widely used for classification and regression tasks in machine learning. The algorithm creates subsets of the original dataset through bootstrapping and constructs decision trees by recursively partitioning the data based on feature splits. The final prediction is determined by aggregating the predictions of all the trees in the ensemble. Random Forest is known for its robustness against overfitting, ability to handle large datasets, and feature importance estimation. However, it can be computationally expensive and less interpretable compared with single decision trees. The hyperparameters with essential values of random forest are shown in Table 2.

```
  Data: Data set D, Original features matrix X,
        total instances in the data set n, target
        instance Rₖ, k is iᵗʰ instance, Target output
        classes label is y, w[i], Target number of
        features "N"
  Result: Reduced feature set S.
1 Begin
2 Step 1: The algorithm takes as input a feature
    matrix "X" of shape "(n samples, n features)." a
    target variable "y" of shape "(n samples)," a
    chosen machine learning model, and a target
    number of features "N";
3 Step 2: It initializes two lists: "selected
    features" to store the indices of the selected
    features and "remaining features" to keep track
    of the indices of the features that have not
    been selected yet;
4 Step 3: The algorithm enters a loop that
    continues until the desired number of features
    "N" is reached;
5 Step 4: Inside the loop, the algorithm trains
    the model on the remaining features, calculates
    the importance scores for each feature, and
    identifies the least important feature;
6 Step 5: The least important feature is appended
    to the "selected features" list and removed from
    the "remaining features" list;
7 Step 6: The model is retrained using the updated
    feature subset;
8 Step 7: The loop continues until the desired
    number of features "N" is obtained;
9 Step 8: Finally, the algorithm returns the
    "selected features" list containing the indices
    of the selected features;
10  Finish;
```

Algorithm 1.   Recursive Feature Elimination (RFE) algorithm.

2.3.2 Ada Boost ensemble learning algorithm

AdaBoost (AB) (22) is an ensemble learning algorithm that puts together weak learners to form a strong classifier. It iteratively trains weak learners on weighted data, focusing on misclassified samples. The resulting prediction is a weighted combination of weak learners' predictions. AdaBoost handles complex decision boundaries and achieves high accuracy but can be sensitive to noise and outliers. The hyperparameters with essential values of Ada Boost algorithm are shown in Table 3.

2.4 Stacking model based on Random Forest and Ada Boost algorithms

The stacking approach is an ensemble technique for training several base classifiers on the same dataset. Instead of making individual predictions, the predictions of these base classifiers are combined using a meta-classifier, which is typically a model such as logistic regression, random forest, or a neural network. The meta-classifier learns to make predictions based on the outputs of the base classifiers. By combining different types of classifiers, each with its strengths and weaknesses, the stacking approach aims to leverage the diverse perspectives and expertise of the individual classifiers to improve overall classification performance. This can lead to higher accuracy and better generalization compared with using a single classifier.

In this study, we trained two base classifiers (Random Forest and Ada Boost) using the entire training set. By using these two techniques, we aimed to introduce more diversity and variation into the ensemble. The predictions of each base model, Random Forest, and Ada Boost are then combined and used to train the meta-classifier, which in this case is also a Random Forest model.

TABLE 2   Random Forest hyperparameters with essential values.

Parameters name	Description	Values
N estimators	Number of decision trees in the forest	1,000
Max depth	Maximum depth of each decision tree	20
Min samples split	Minimum number of samples required to split an internal node	10
Min samples leaf	Minimum number of samples required to be at a leaf node	5
Max features	Maximum features to use for splitting at each node	Randomness FS
Bootstrap	A boolean indicating whether to use bootstrap samples for training	True or False
Criterion	Function to measure the quality of a split (e.g., Gini impurity, entropy)	Entropy
Class weight	Weights associated with each class in classification tasks to handle class imbalance	Balance
Random state	Random seed for reproducibility	None

TABLE 3   Ada Boost algorithm hyperparameters with essential values.

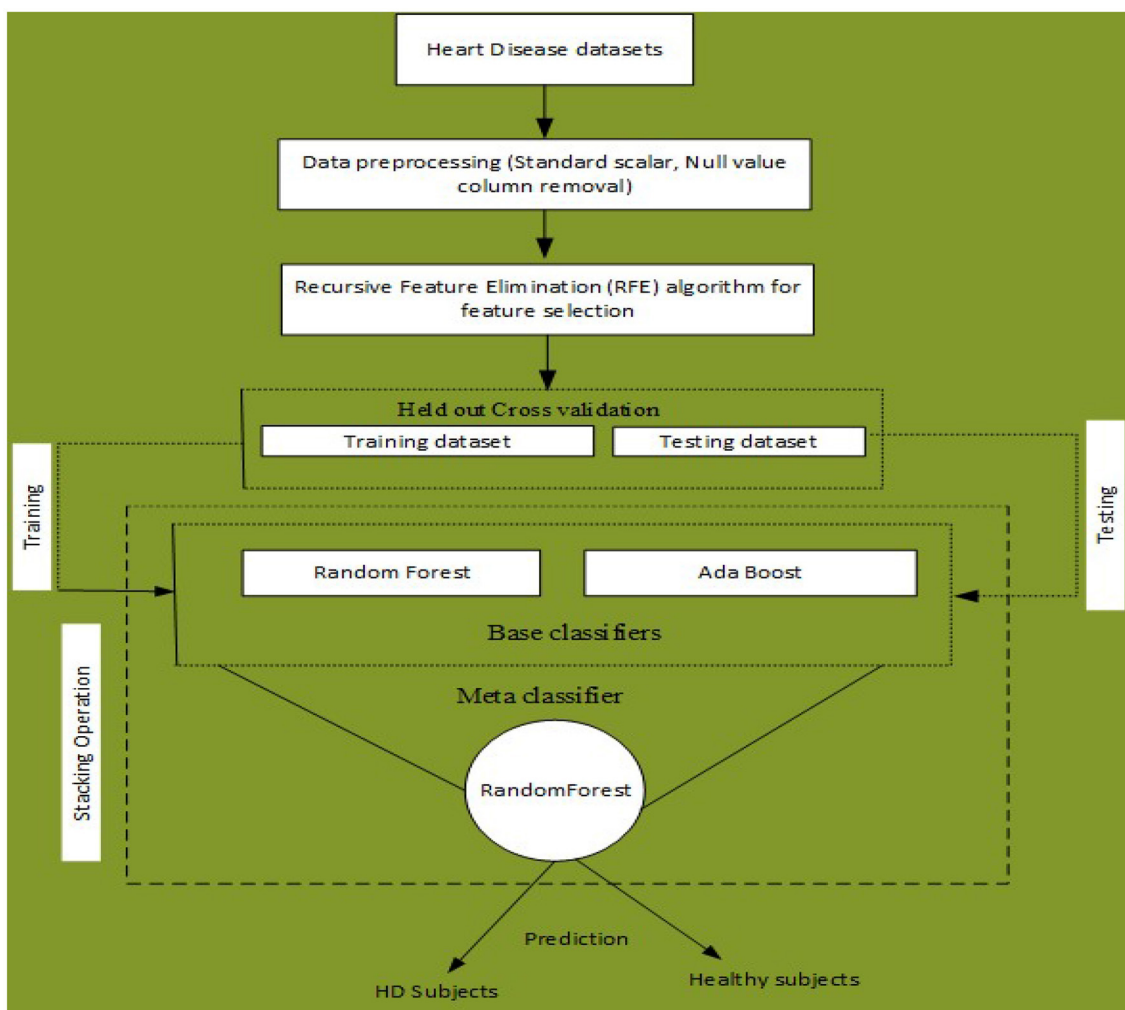| Parameters name	Description	Values
N estimators	Parameters determine the number of weak learners to be included in the ensemble	200
Learning rate	Controls the contribution of each weak learner to the final prediction	0.001
Base estimator	Parameter specifies the weak learner used in the ensemble	–
Algorithm	Determines the algorithm used to update sample weights during training	"SAMME.R"

FIGURE 1
Proposed stacking-based (Stacking HD) model for Heart disease diagnosis in IoT healthcare systems.

2.5 Model cross validation

The model was trained and validated using the held-out cross-validation procedure (2). When the data set is large, the holdout CV is an appropriate validation approach. In this study, heart disease datasets such as CHD, CHDP, and SCH data sets were used and separated into 80% for training and 20% for model testing.

2.6 Performance evaluation criteria

The performance evaluation metrics (6) were used in this study to evaluate the proposed model performance. These evaluation metrics were expressed in equations mathematically Equations 1–6, respectively. TP denotes True Positive, TN denotes True Negative, FP denotes False Positive, and FN is False Negative.

$$\textbf{Accuracy (Acc)} = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \qquad (1)$$

$$\textbf{Sensitivity (Sn)} = \frac{TP}{TP + FN} \times 100 \qquad (2)$$

$$\textbf{Specificity (Sp)} = \frac{TN}{TN + FP} \times 100 \qquad (3)$$

$$\textbf{Precision (Pr)} = \frac{TP}{TP + FP} \times 100 \qquad (4)$$

$$\textbf{F1-Score (F1-S)} = 2 \times \frac{Pr \times \text{Recall}}{Pr + \text{Recall}} \times 100 \qquad (5)$$

Matthews correlation coefficient (MCC):

$$MCC = \frac{T_1}{\sqrt{T_2 \times T_3 \times T_4 \times T_5}} \times 100 \qquad (6)$$

where $T_1 = TP \times TN - FP \times FN$, $T_2 = TP + FP$, $T_3 = TP + FN$, $T_4 = TN + FP$, and $T_5 = TN + FN$.

```
    Input : Cleveland Heart Disease dataset (CHD),
            Heart Statlog Cleveland Hungary dataset
            (HeartStatlogClevelandHungary)
    Output: Evaluation metrics for all features and
            selected features
 1  Step 1: Data preparation;
 2  StandardizeData dataset return
    StandardScale(dataset);
 3  CHD ← LoadDataset("Cleveland Heart Disease
    dataset");
 4  HeartStatlogClevelandHungary ←
    LoadDataset("Heart Statlog Cleveland Hungary
    dataset");
 5  StandardizedCHD ← StandardizeData(CHD);
 6  StandardizedHeartStatlog ←
    StandardizeData(HeartStatlogClevelandHungary);
 7  TrainData, TestData ←
    HoldOutSplit(StandardizedCHD);
 8  Step 2: Model construction;
 9  TrainRandomForest data, num_trees
    RandomForestModel ← RandomForest(num_trees);
10  RandomForestModel.fit(data);
11  return RandomForestModel;
12  TrainAdaBoost data, num_trees AdaBoostModel ←
    AdaBoost(num_trees);
13  AdaBoostModel.fit(data);
14  return AdaBoostModel;
15  RFE data return RecursiveFeatureElimination(data);
16  RandomForestModel ← TrainRandomForest(TrainData,
    num_trees=1000);
17  AdaBoostModel ← TrainAdaBoost(TrainData,
    num_trees=200);
18  SelectedFeatures ← RFE(TrainData);
19  Stacking;
20  StackingModelPredict data, base Models
    BaseModelOutputs ← [];
21  for model in baseModels do
22  │   BaseModelOutputs.append(model.predict(data));
23  end
24  StackingModelInput ←
    Concatenate(BaseModelOutputs);
25  StackingModelOutput ←
    RandomForestPredict(StackingModel,
    StackingModelInput);
26  return StackingModelOutput;
27  BaseModelOutputsAllFeatures ←
    StackingModelPredict(TestData,
    [RandomForestModel, AdaBoostModel]);
28  RandomForestModelSelectedFeatures ←
    TrainRandomForest(TrainData[SelectedFeatures],
    num_trees=1000);
29  AdaBoostModelSelectedFeatures ←
    TrainAdaBoost(TrainData[SelectedFeatures],
    num_trees=200);
30  BaseModelOutputsSelectedFeatures ←
    StackingModelPredict(TestData[SelectedFeatures],
    [RandomForestModelSelectedFeatures,
    AdaBoostModelSelectedFeatures]);
```

```
31  Step 3: Model performance evaluation;
32  EvaluationMetricsAllFeatures ←
    EvaluateModel(BaseModelOutputsAllFeatures,
    TestData);
33  EvaluationMetricsSelectedFeatures ←
    EvaluateModel(BaseModelOutputsSelectedFeatures,
    TestData);
```

Algorithm 2. Stacking HD heart disease diagnosis.

Area under the ROC curve AUC:

The AUC represents the model's ROC, and a high AUC number indicates a high-performance model. These equations represent various performance metrics commonly used in binary classification tasks.

2.7 Proposed model (stacking HD)

An ML-based computer-aided diagnosis (CAD) model for detecting HD early stages in the Medical Internet of Things (IoT) system. In the designing of the CAD system, we applied data pre-processing techniques such as standard scalar and the removal of null values from the data set. To select related features from the data set, we incorporated the Recursive Feature Elimination (RFE) algorithm. This helps to balance the data for proper training of the algorithm and enhance the algorithm's predictive capability. The machine learning classifiers Random Forest (RF) and Ada Boost (AB) were used for the classification of affected and healthy control subjects. These models were trained and evaluated using the entire data set and selected feature data set. To further improve the predictive results of these models, we incorporated a stacking approach to select the best meta-classifier between the Random Forest and Ada Boost. We defined a parameter grid for grid search for both algorithms. Furthermore, a hold-out validation mechanism was utilized and data were split for training and testing in portions of 80 and 20%, respectively. The Cleveland Heart Database was used to validate the proposed model. Different performance assessment metrics were computed for model evaluation. The experimental results unequivocally demonstrated that our proposed model outperformed the baseline models in terms of predictive accuracy. The model flowchart is shown in Figure 1, and the model's method in Algorithm 2 is as follows.

3 Experiments

3.1 Experiments setup

For the implementation of the proposed model, we performed various experiments. First, we incorporated data preprocessing and feature selection techniques to balance the data set and remove the irrelevant features from the data set. The ML classifiers Random Forest and Ada Boost were trained on 80% the original feature data set and the selected feature data set and evaluated with 20% data.

Furthermore, as shown in Tables 2, 3, additional hyperparameters were adjusted in each model accordingly. The Cleveland Heart Disease and Heart Statlog Cleveland Hungary datasets were used for validation of the models. To further improve the predictive performance, a stacking mechanism was used.

The proposed model performance was evaluated by computing various evaluation metrics. The experiments were carried out on a laptop and run with a Google collaborator accelerator. All experiments required Python v3.7 and other machine-learning libraries. Consistent values are obtained after repeating the experiments several times. The results of all experiments were provided in tables and graphed.

3.2 Results and analysis

3.2.1 Results of data pre-processing

On the Cleveland Heart Disease dataset (CHD), the proposed model was tested. The original data set has 303 records and 75 columns; however, all published studies used only 14 columns. We did pre-processing on the data set, and 6 records were discarded due to empty values. Hence, the dataset has 297 records with 13 columns and 1 output column. As a result, a features matrix of 297∗13 is created. We also employed a standard scalar to verify that each feature has a mean of 0 and a variance of 1; consequently, all features have the same coefficient. Furthermore, we duplicated 297 samples three times to increase the size of the data set. The number of samples in the new data set is 3∗297 = 891. As a result, the new dataset, known as the Cleveland Heart Disease Proceeded (CHDP) data set, has a matrix size of 891∗13. The description of the CHD is shown in Table 4.

For cross-validation of the models, we incorporated the data set Heart Statlog Cleveland Hungary (SCH). This dataset has 1,190 samples with 11 columns. These datasets were collected and put in one place to enhance research on CAD-related machine learning and data mining methods and perhaps eventually advance clinical diagnosis and early treatment. The feature set Statlog Cleveland Hungary data set is shown in Table 5. The models were trained with Cleveland Heart Disease of feature matrix dataset 297∗13 and 3∗297 = 891 and tested with Heart Statlog Cleveland Hungary data set.

3.2.2 Results of REF algorithm and feature ranking and selected feature subsets from CHD and SCH data sets

To choose the optimal collection of features from the SCH and CHD data sets, the REF FS method was utilized. Table 6 shows the feature rating and selected feature sets. According to Table 6, these feature sets have a significant influence on the classification of HD and HC control subjects. From CHD data set, the subset of selected features included SEX, CTP, EIA, PES, and VCA. While from SCH data set, the selected subset of features are SEx, CTP, FBG, EIA, and PES. We have performed experiments on full and selected feature datasets of both data sets in the coming sections in order to check the models' results on full and selected feature sets.

TABLE 4 Description of cleveland heart disease (CHD) dataset (features matrix of 297 * 13).

Feature name	Feature code	Feature description
Age	AGE	Age in years
Sex	SEX	Male = 1 and Female = 0
Chest pain	CTP	Atypicalangina = 1, Typicalangina = 2, Asymptomatic = 3, Nonanginalpain = 4
Resting blood pressure	RBP	mm hg, hospitalized
Serum cholesterol	SCH	In mg/dl
Fasting blood sugar >120 mg/dl	FBS	fasting blood sugar >120 mg/dl ($T = 1$, $F = 0$)
Resting electrocardiographic	RES	Normal = 0, STT = 1, Hypertropy = 2
Maximum heart rate	MHR	–
Exercise included angina	EIA	Yes = 1, No = 0
Old peak = ST depression included by exercise relative to rest	OPK	–
Slope peak exercise St segment	PES	Up sloping = 1, Flat = 2, Down sloping = 3
Number of major vessels (0–3) colored by fluoroscopy	VCA	–
Thallium scan	THA	Normal = 3, Fixed defect = 6, Reversible defect = 7
Lable	LB	Heart disease = 1, healthy = 0

TABLE 5 Description of Statlog Cleveland Hungary (SCH) data set (features matrix of 1,190 * 11).

Feature name	Feature code	Feature description
Age	AGE	Age in years
Sex	SEX	Male = 1 and female = 0
Chest pain	CTP	Atypical-angina = 1, typical-angina = 2, Asymptomatic = 3, Non-anginal-pain = 4
Resting blood pressure	RBP	mm hg, hospitalized
Serum cholesterol	SCH	In mg/dl
Fasting blood sugar >120 mg/dl	FBS	fasting blood sugar >120 mg/dl ($T = 1$, $F = 0$)
Resting electrocardiographic	RES	Normal = 0, STT = 1, Hypertropy=2
Maximum heart rate	MHR	–
Exercise included angina	EIA	Yes = 1, No = 0
Old peak = ST depression included by exercise relative to rest	OPK	–
Slope peak exercise St segment	PES	Up sloping = 1, flat = 2, down sloping = 3
Targrt	TG	Heart disease = 1, healthy = 0

TABLE 6 Feature ranking and selected feature subsets from CHD and SCH data sets by REF algorithm, i.e., $297 * 5 \subset 297 * 13$ and $1,190 * 5 \subset 1,190 * 11$.

Dataset	Feature name	Feature code	Feature ranking	Selected feature
CHD	Age	AGE	7	
	Sex	SEX	1	SEX
	Chestpain	CTP	1	CTP
	Resting blood pressure	RBG	8	
	Serum cholesterol	SCH	9	
	Fasting blood sugar	FBG	2	
	Resting electrocardiographic	RES	5	
	Maximum heart rate	MHR	6	
	Exercise included angina	EIA	1	EIA
	OldPeak	OPK	3	
	SlopofST	PES	1	PES
	Flouroscorpy	VCA	1	VCA
	Thal	THA	4	
SCH	Age	AGE	6	
	Sex	SEX	1	SEX
	Chestpain	CTP	1	CTP
	Resting blood pressure	RBG	5	
	Serum cholesterol	SCH	7	
	Fasting blood sugar	FBG	1	FBG
	Resting electrocardiographic	RES	3	
	Maximum heart rate	MHR	4	
	Exercise included angina	EIA	1	EIA
	OldPeak	OPK	4	
	SlopofST	PES	1	PES

3.2.3 Results of Random Forest and Ada Boost with full and selected feature data sets

The classification performance of Random Forest and Ada Boost was evaluated on whole and selected feature datasets of CHD, CHDP, and SCH datasets, respectively. The models were configured with basic hyperparameters, as shown in Tables 2, 3. The held-out cross-validation was incorporated, and data sets were divided into 80 and 20% ratios for training and validating of the models, respectively. The model's performance was evaluated by computing different evaluation metrics, and the results were reported and discussed in detail.

Table 7 presented the results of classifiers Random forest and Ada boost trained and evaluated on full and selected feature sets on the CHD data set. On the full feature set, obtained results are 88.33% accuracy, 88.45% specificity, 89.23% sensitivity, 94.65% precision, 91.02% MCC, and 89.02% F1-score. While on selected features set the model 89.12%, 92.24%, 88.22%, 89.98%, 93.24%, and 90.00%, respectively. The model improved accuracy 89.12–88.33 = 0.79% on the selected feature set. The performance of other metrics also

greatly improved. In Figure 2, Random Forest results are graphically presented.

The Ada Boost results are presented in Table 7 with the full feature set and obtained 78.33%, 78.21%, 92.11%, 89.34%, 91.00%, and 79.21% of accuracy, specificity, sensitivity, precision, MCC, and F1-score, respectively. On the selected feature set, the Ada Boost achieved 78.78%, 97.23%, 88.65%, 93.36%, 92.02%, and 80.58% of accuracy, sensitivity, specificity, precision, MCC, and F1-score values, respectively. Figure 3 graphically presents the model results of Ada boost on both selected and full feature data sets of CHD data set.

Table 8 presented the results of classifiers Random forest and Ada boost trained and evaluated on full and selected feature sets on the CHDP data set. The accuracy, specificity, sensitivity, precision, MCC, and F1-score values on the full feature set were 98.34%, 98.45%, 98.32%, 93.67%, 97.33%, and 98.32%, while those values on selected feature set were 98.89%, 99.00%, 98.77%, 98.67%, 96.00%, and 99.01%, respectively. The model improved accuracy 98.89–98.34 = 0.54% on the selected feature set. The performance of other metrics also greatly improved. In Figure 4, Random Forest results are graphically presented.

TABLE 7 Results of Random Forest and Ada Boost with full and selected feature sets of CHD data set.

Model	Data set	Metrics					
		Acc (%)	Sp (%)	Sn (%)	Pr (%)	MCC (%)	F1-S (%)
Random forest	Full feature	88.33	88.45	89.23	94.65	91.02	89.02
–	Selected feature	89.12	92.24	88.22	89.98	93.24	90.00
Ada boost	Full feature	78.33	78.21	92.11	89.34	91.00	79.21
–	Selected feature	78.78	97.23	88.65	93.36	92.02	80.58
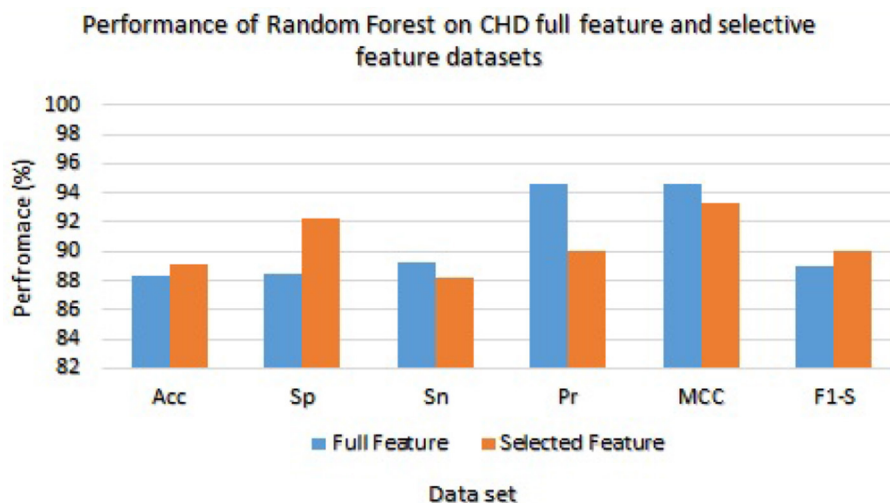
FIGURE 2
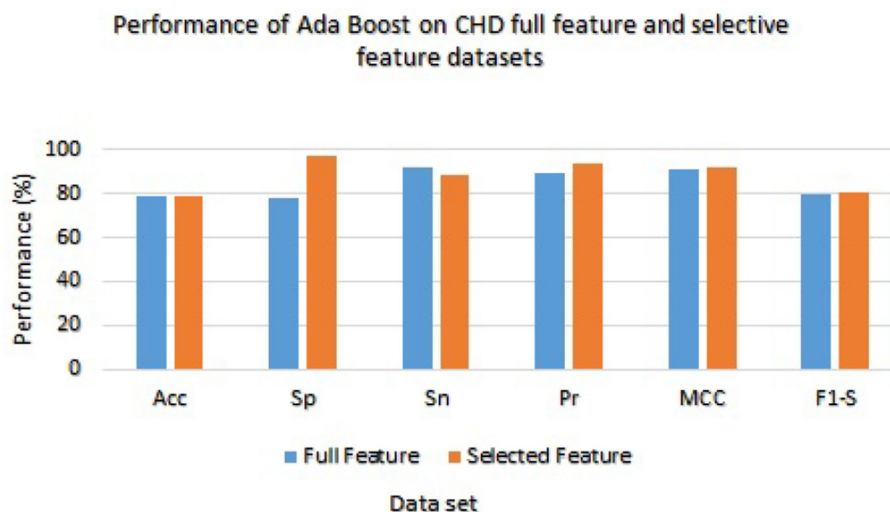Results of Random Forest with full and selected feature sets of (CHD) data set.

FIGURE 3
Results of Ada Boost with full and selected feature sets of (CHD) data set.

On the other hand, Ada Boost results with CHDP dataset are presented in Table 8 with the full feature set and obtained 93.29% accuracy, 93.28% specificity, 93.02% sensitivity, 94.00% precision, 93.89% MCC, and 94.02% F1-score. The Ada Boost achieved 93.89% accuracy, 93.89% specificity, 94.09% sensitivity, 95.09% precision, 94.23% MCC, and 94.43% F1-measure on the specified feature set. Figure 5 graphically presented the model results of Ada boost

TABLE 8 Results of Random Forest and Ada Boost with full and selected feature sets of CHDP data set.

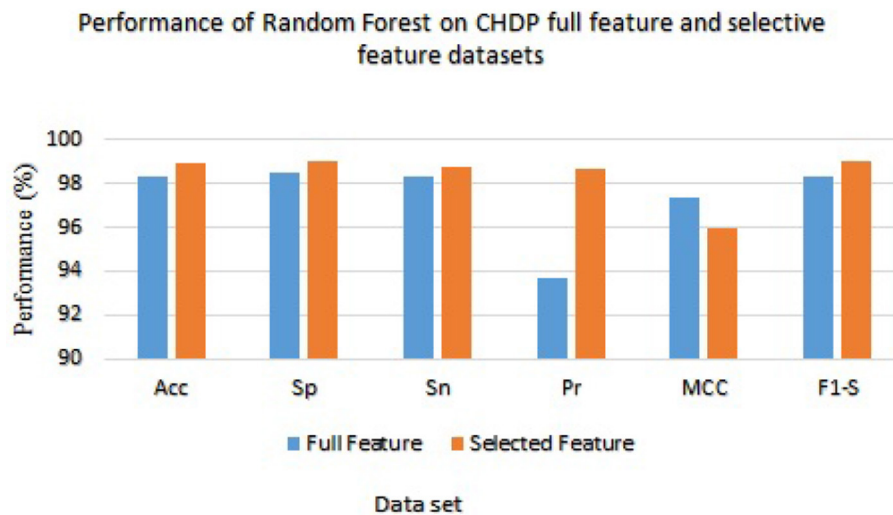| Model	Dataset	Metrics					
		Acc (%)	Sp (%)	Sn (%)	Pr (%)	MCC (%)	F1-S (%)
Random forest	Full feature	98.34	98.45	98.32	93.67	97.33	98.32
–	Selected feature	98.89	99.00	98.77	98.67	96.00	99.01
Ada boost	Full feature	93.29	93.28	93.02	94.00	93.89	94.02
–	Selected feature	93.89	93.99	94.09	95.09	96.23	94.43

FIGURE 4
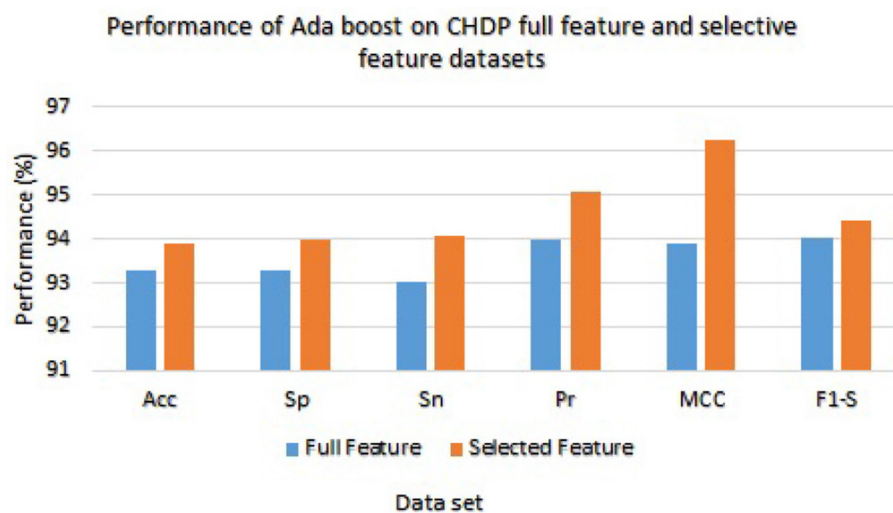Results of Random Forest with full and selected feature sets of CHDP data set.

FIGURE 5
Results of Ada Boost with full and selected feature sets of CHDP data set.

on both selected and full feature data sets of the CHDP data set.

We have checked the model's performance on full and selected feature data sets (SCH) in order to evaluate these models.

Table 9 presented the Random Forest and Ada Boost classifier's experimental results. With the full feature set, the Random Forest gained 94.53%, 94.59%, 94.56%, 95.02%, 94.33%, and 94.53% of accuracy, specificity, sensitivity, precision, MCC, and F1-score,

TABLE 9 Results of Random Forest and Ada Boost with full and selected feature sets of SCH data set.

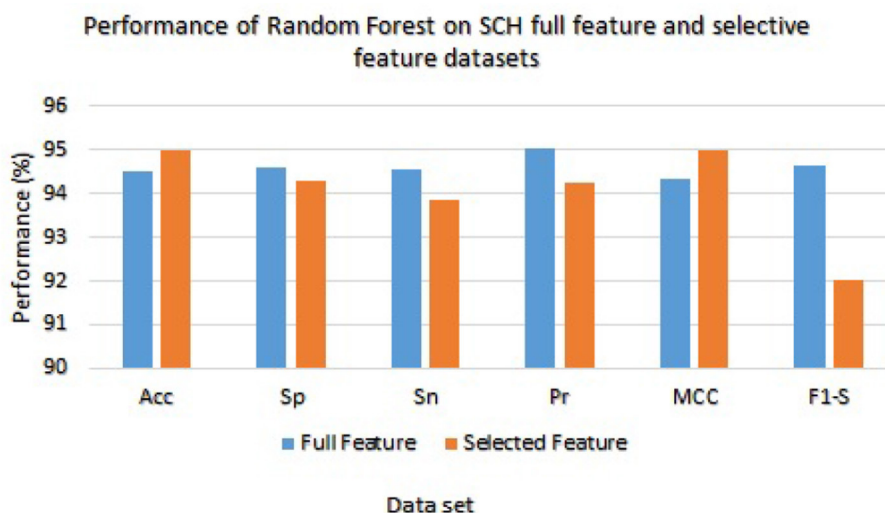| Model	Dataset	Metrics					
		Acc (%)	Sp (%)	Sn (%)	Pr (%)	MCC (%)	F1-S (%)
Random forest	Full feature	94.53	94.59	94.53	95.02	94.33	94.53
–	Selected feature	95.00	94.30	93.87	94.23	95.01	92.04
Ada boost	Full feature	86.96	86.98	86.89	97.92	86.00	87.00
–	Selected feature	87.02	98.99	86.23	87.36	88.98	87.98

FIGURE 6
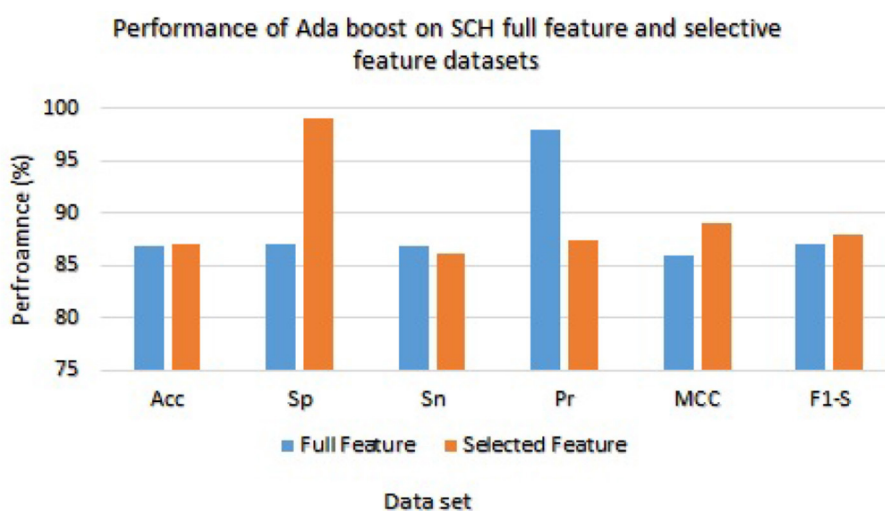Results of Random Forest with full and selected feature sets of (SCH) data set.

FIGURE 7
Results of Ada Boost with Full and Selected Feature sets of SCH data set.

respectively. While accuracy, sensitivity, specificity, precision, MCC, and F1-score values on the selected feature set the Random Forest achieved 95.00%, 94.30%, 93.87%, 94.23%, 95.01%, and 92.04%, respectively. Figure 6 graphically presented the model results of Random Forest on both selected and full feature data sets of the SCH data set.

The Ada Boost results on full and selected feature data sets (SCH) are shown in Table 9. On the full feature set, the
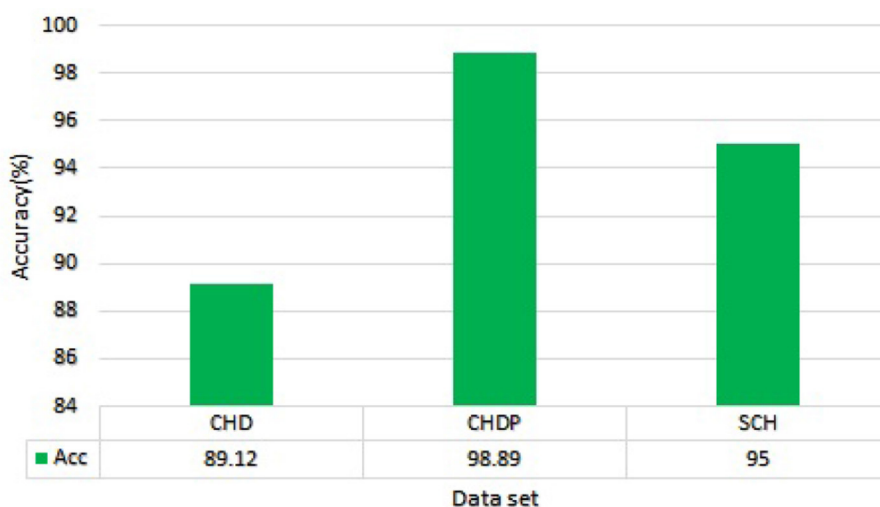
FIGURE 8
Accuracy comparison of Random Forest on three data sets.

TABLE 10 Classifier evaluation with cross dataset.

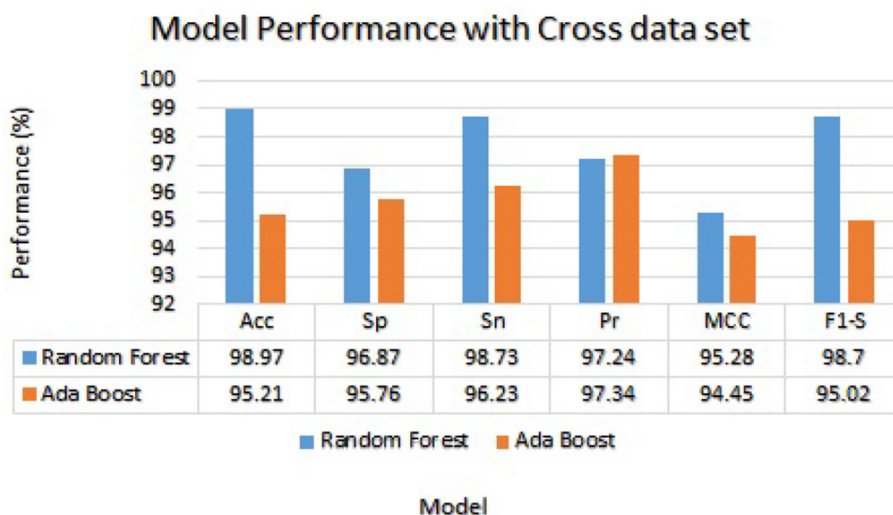| Model	Metrics					
	Acc (%)	Sp (%)	Sn (%)	Pr (%)	MCC (%)	F1-S (%)
Random Forest	98.97	96.87	98.73	97.24	95.28	98.70
Ada Boost	95.21	95.76	96.23	97.34	94.45	95.02

FIGURE 9
Model results trained and validated with the independent cross-data set.

Ada boost achieved 86.96% accuracy, 86.98% specificity, 86.89% sensitivity, 97.92% precision, 86.00% MCC, and 87.00% F1-score. The Ada Boost improved predictive performance on selected feature dataset and obtained 87.02% accuracy, 98.99% specificity, 86.23% sensitivity, 87.36% precision, 88.98% MCC, and 87.98% F1-score. Figure 7 graphically displayed the Ada Boost model results on both the selected and full feature data sets of the SCH data set.

On the basis of the experimental results of Random Forest and Ada Boost classifiers on full and selected feature sets on three datasets including, CHD, CHDP, and SCH, as shown in Tables 7–9, we concluded that the performance of Random Forest algorithm is

TABLE 11 Stacking HD model performance with CHD, CHDP, and SCH data sets.

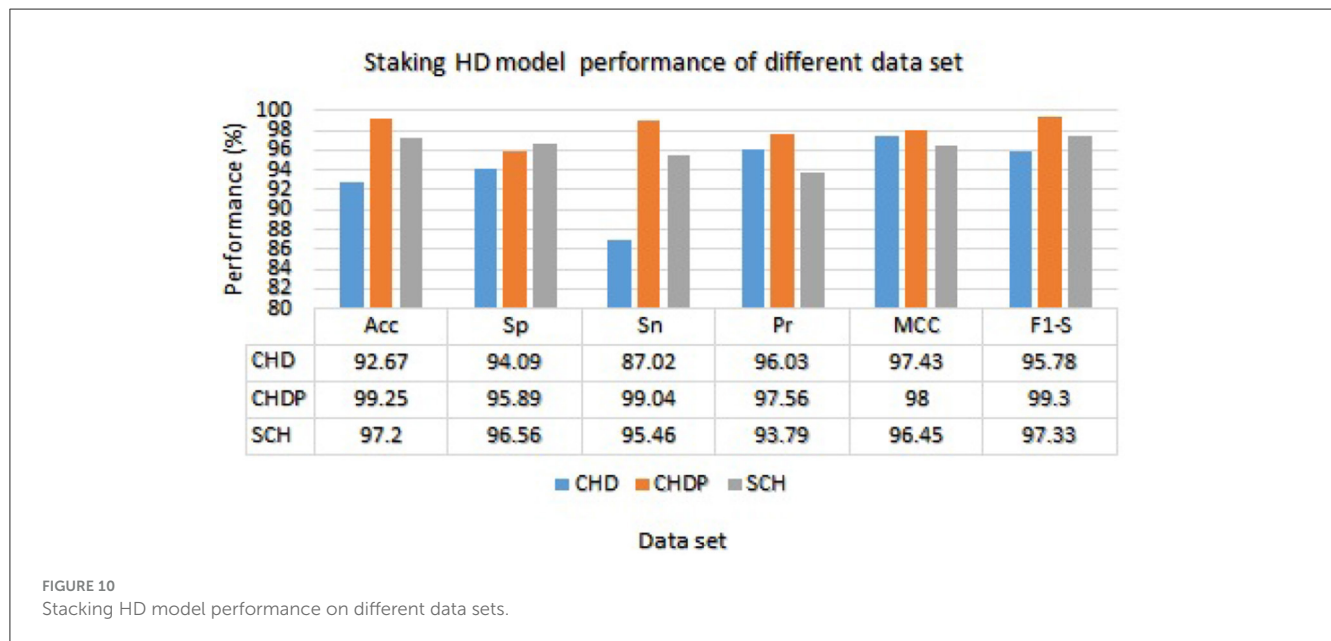| Model	Datasets	Acc (%	Sp (%)	Sn	Pr (%)	MCC (%)	F1-S (%)
Random Forest	CHD	92.67	94.09	87.02	96.03	97.43	95.78
Random Forest	CHDP	99.25	95.89	99.04	97.56	98.00	99.30
Random Forest	SCH	97.20	96.56	95.46	93.79	96.45	97.33

FIGURE 10
Stacking HD model performance on different data sets.

higher as compared with Ada Boost algorithm on CHDP data set. In terms of accuracy, Random forest with CHDP data set obtained 98.89% classification accuracy. On CHD data set, the accuracy of RF algorithm was 89.12% and the accuracy of SCH data set was 95.00%. Thus, on the basis of the data set, the Random forest classifier in CHDP data set is higher than in CHD and SCH data sets. Hence, Random Forest is a suitable classifier for the diagnosis of HD in IoT healthcare systems. The RF performance in terms of accuracy on three data sets is graphically presented in Figure 8 for better understanding.

3.2.4 Models performance evaluation with cross dataset

With separate cross-datasets, we examined the predictive outcomes of the Random Forest (RF) and Ada Boost (AB) classifiers. We trained the Random Forest and Ada Boost with CHD data set and tested with an independent SCH data set. The models were configured with basic hyperparameters as shown in Tables 2, 3. The model's performance was evaluated by computing different evaluation metrics and experimental results, as shown in Table 10.
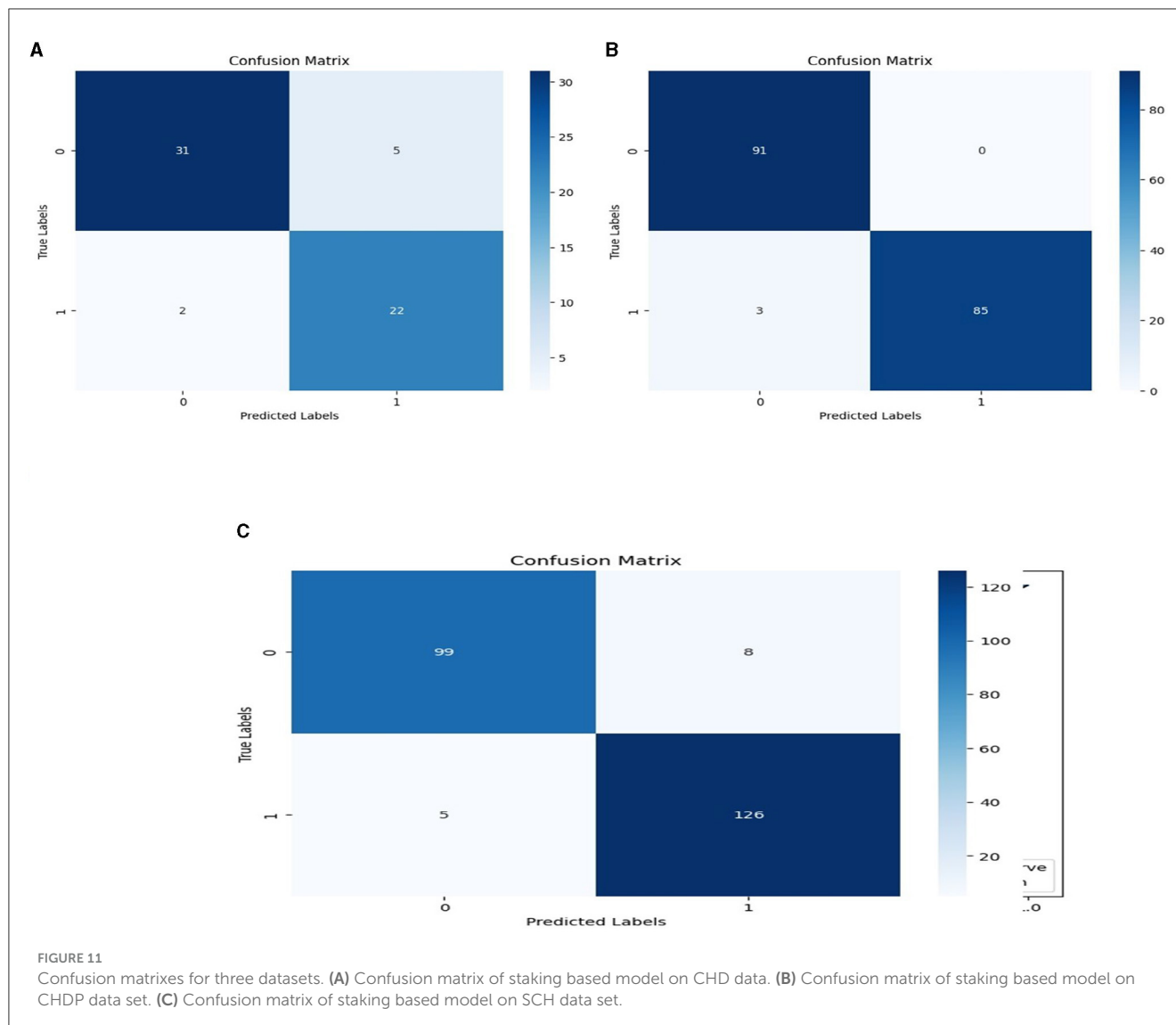
Table 10 reported performance metrics results for the random forest model including accuracy, sensitivity, specificity, precision, MCC, and F1-score which were 98.97%, 96.87%, 98.73%, 97.24%, 95.28%, and 98.70%, respectively. The test accuracy of the Random forest model is higher as compared to the test accuracy of the Ada Boost model on the same data. While the Ada Boost reached an accuracy of 95.21%, a specificity of 95.76%, a sensitivity of 96.23%,

a precision of 97.34%, MCC of 94.45%, and F1-score of 95.02%. The test accuracy is higher as compared to the test accuracy of the same data. The cross-data performance of Random Forest and Ada Boost is graphically shown in Figure 9.

3.2.5 Results of the stacking model (stacking HD)

We used the performance of all models (Random Forest and Ada Boost) as new training data to increase classification performance. The Random Forest model results were highest between Random Forest and Ada Boost models when the selected feature data sets of CHD, CHDP, and SCH were used. The outcomes of the stacking-based model (stacking HD) are shown in Table 11. The stacking-based model (stacking HD) performance of different data sets is presented graphically in Figure 10 for better understanding. The table presents that the results of the stacking-based model (stacking HD) are better and obtained 92.67% accuracy, 94.09% specificity, 87.02% sensitivity, 96.03% precision, 97.43% MCC, and 95.78% F1-score on the CHD selected feature data set. The performance of the stacking approach on CHD data is better than that of individuals models Random forest as reported in Table 7 such as 89.12% accuracy, 92.24% specificity, 88.22% sensitivity, 89.98% precision, 93.24% MCC and 90.00% F1-score. The Confusion Matrix (CM) and ROC curve of the stacking-based model on CHD data set are shown graphically in Figures 11A, 12A.

While on CHDP selected feature dataset, the stacking HD model meta classifier (Random Forest) obtained the higher performance in terms of 99.25% accuracy, 95.89% specificity,

FIGURE 11
Confusion matrixes for three datasets. **(A)** Confusion matrix of staking based model on CHD data. **(B)** Confusion matrix of staking based model on CHDP data set. **(C)** Confusion matrix of staking based model on SCH data set.
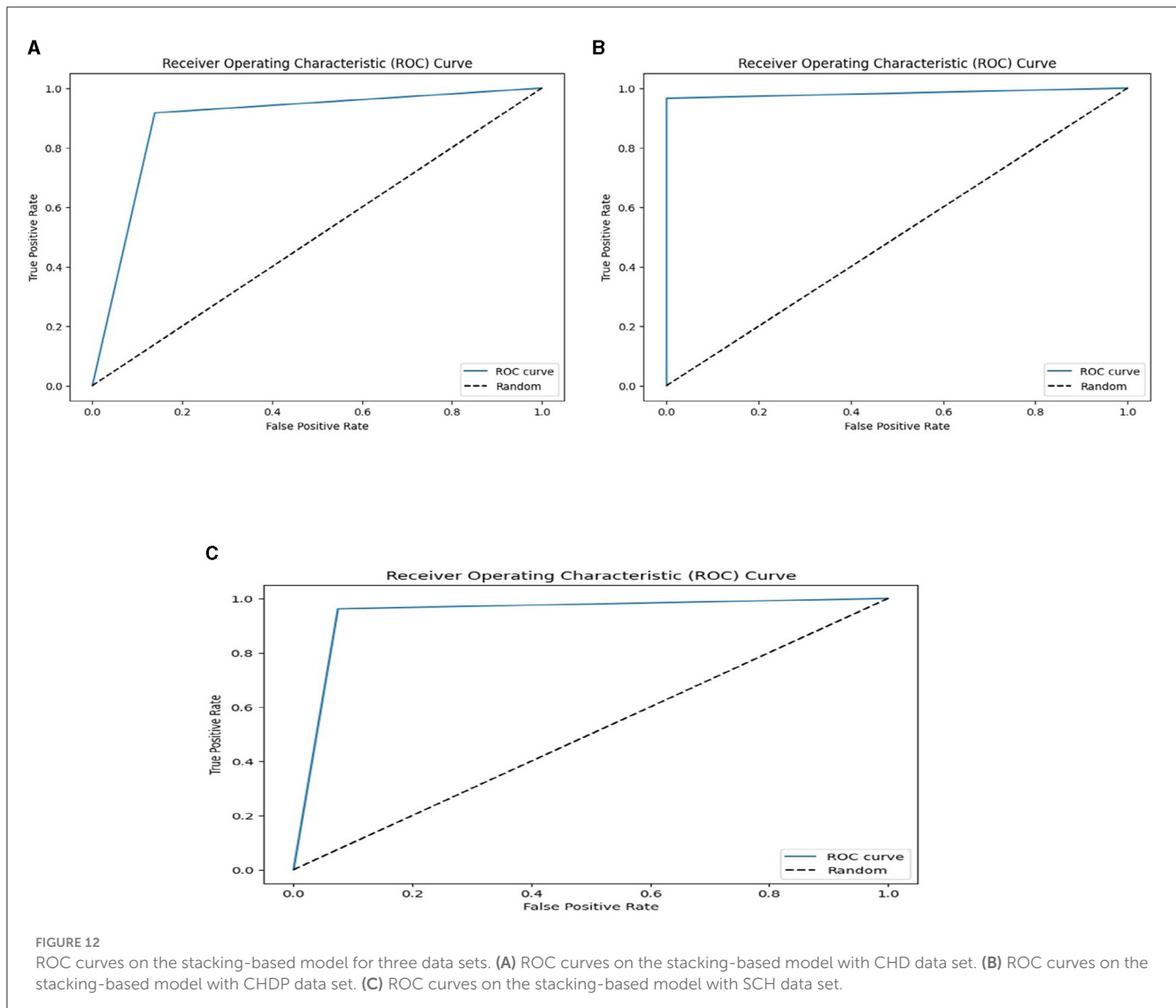
99.04% sensitivity, 97.56% precision, 98.00% MCC, and 99.30% F1-measure. The CM and ROC curves of the stacking-based model on CHDP data set are shown graphically in Figures 11B, 12B. The stacking approach-based model on the SCH data set obtained 97.20% accuracy, 96.56% specificity, 95.46% sensitivity, 93.79% precision, 96.45% MCC, and 97.33% F1-score. The CM and ROC curves of the stacking-based model on SCH data set are shown graphically in Figures 11C, 12C. The above stacking-based model (Stacking HD) results on different data sets presented that stacking-based models perform better than individual models. The result of the stacking-based model is the high performance of CHDP data set as compared with CHD and SCH data sets. Among the three stacking model, the stacking HD on the CHDP data set obtained a higher accuracy of 99.25%. Hence, the stacking HD model is an appropriate method to diagnose HD in its early stages. Random forest is considered as the meta classifier.

3.2.6 Comparison of stacking HD model with existing models

The proposed model (stacking HD) predictive accuracy is compared with baseline models, as shown in Table 12. Table 12 presented that the stacking HD model reached a higher 99.25% accuracy as compared with baseline models. The suggested method's great performance revealed that it correctly diagnoses HD and may be simply applied in IoT healthcare for the diagnosis of heart diseases.

4 Discussion

The diagnosis of heart disease (HD) is a critical task in the early stages of IoT healthcare systems. World Health Organization (WHO) reported that a large number of people are suffered from HD each year (1). To handle the initial stages of recognition of HD,

FIGURE 12
ROC curves on the stacking-based model for three data sets. **(A)** ROC curves on the stacking-based model with CHD data set. **(B)** ROC curves on the stacking-based model with CHDP data set. **(C)** ROC curves on the stacking-based model with SCH data set.

various diagnosis methods have been proposed by medical experts and researchers. Machine learning techniques based on Computer-Aided Diagnostic Systems (CAD) in an IoT healthcare system can accurately detect HD in its initial phases (30, 31). Machine learning techniques are widely used in CAD systems to diagnose critical diseases such as heart disease in IoT healthcare (32, 33). However, the existing HD diagnostic methods have the problem of lack of accuracy in the diagnosis HD correctly. The low prediction accuracy arises due to imbalanced data and irrelevant feature data for the ML model training. To address this issue, a new approach for properly and efficiently diagnosing heart disease is required for IoT healthcare systems.

The research study designed machine learning technique-based CAD systems for HD diagnosis in IoT-healthcare systems. In the designing of the CAD system, data pre-processing techniques such as standard scalar and removing null values attribute records from the data set. For related feature selection from the data set, we incorporated the Recursive Feature Elimination (RFE) algorithm to balance the data for good training of the model to enhance the model's predictive capability. The machine learning classifiers

Random Forest and Ada Boost were used for the classification of affected and healthy control subjects. These models were trained and evaluated using the entire data set and selected feature data set. To further improve the predictive performance of these models, we incorporated a stacking approach to select the best meta-classifier between the Random Forest and Ada Boost. We defined a parameter grid for grid search for both algorithms.

Furthermore, the held-out validation procedure was used, and data were split into sections of 80 and 20% for training and testing. The proposed model was validated using CHD, CHDP, and SCH databases. For model performance evaluation, various performance assessment metrics results were generated. The experimental results were compared with the existing state of the arts methods.

Here, the experimental results are briefly presented. The RFE algorithm from the CHD data set of the subset of selected features included SEX, CTP, EIA, PES, and VCA. While from the SCH data set, the selected subsets of features are SEX, CTP, FBG, EIA, and PES. The performance of the Radom Forest algorithm on CHDP data was higher as compared with CHD and SCH data sets. Hence, Table 8 presented the results of the

TABLE 12 Proposed model performance comparison with baseline models.

Model	Acc (%)	Ref
ML algorithms	77	(7)
MLP + SVM	80.41	(16)
Random Forest model	88	(23)
GA-RBF	94.20	(24)
Hybrid MLmodel (HRFLM)	88.07	(13)
SGD	87.69	(25)
ANN + Fuzzy Logic (ANN-FL)	87.4	(8)
ANN ensemble-based diagnosis system	89.01	(17)
IHDPS	88.12	(9)
3p-ANN	88.89	(10)
RF	96.72	(26)
ANN-FUZZY-AHP	91.1	(11)
SVM	96.72	(27)
KNN	90.789	(28)
Random Forest	92.3	(29)
RRS-HD	92.32	(12)
HISFP	89	(2)
FCMIM-SVM	92.37	(14)
SE	92.34	(15)
Proposed stacking HD (CHD)	92.67	2024
Proposed stacking HD (CHDP)	99.25	2024
Proposed stacking HD (SCH)	97.20	2024

classifier Random forest trained and evaluated on full and selected feature sets on the CHDP data set. The values for the whole feature set's accuracy, sensitivity, specificity, precision, MCC, and F1-score were 98.34%, 98.45%, 98.32%, 93.67%, 97.33%, and 98.32%. While the values on selected feature set models were 98.89%, 99.00%, 98.77%, 98.67%, 96.00%, and 99.01%, respectively. The model improved accuracy 98.89–98.34 = 0.54% on the selected feature set. The performance of other metrics also greatly improved. The Random Forest accuracy is also higher than the Ada Boost classifier. Similarly, when stacking techniques were incorporated, the Random Forest performance was higher than Ada Boost, and the Random Forest model was selected as the meta-model. According to Table 7, on the CHDP chosen feature dataset, the stacking technique selected the Random Forest meta classifier and produced the higher performance in terms of accuracy, sensitivity, specificity, precision, MCC, and F1-score, each with a score of 99.25%, 95.89%, 99.04%, 97.56%, 98.00%, and 99.30%.

The confusion matrix and ROC curves of the stacking approach with data sets CHD, CHDP, and SCH are shown in Figures 11, 12. Hence, the ROC curve of the stacking model with the CHDP data set is higher, so it presents that the model accurately detected the HD as compared with CHD and SCH data sets.

Our analysis of the aforementioned results led us to the conclusion that the proposed model, stacking HD, provided better predictive outcomes and was easily implementable for HD detection in IoT-based healthcare systems.

5 Conclusion and future work direction

Machine learning-based Computer-Aided Diagnosis Systems are typically utilized to effectively identify heart disease. However, because current artificial diagnostic approaches are imprecise, medical practitioners are not adopting them into the heart diagnosis process efficiently. In the research study, we created an accurate technique for identifying HD using ML techniques. In the proposed approach, machine learning classifiers including Random-Forest (RF) and Ada-Boost are incorporated for the classification of heart disease and healthy control subjects. For data pre-processing and feature selection, we incorporated standard scalar and Recursive Feature Elimination (RFE) techniques to balance the data for proper training of the algorithm to enhance the model's predictive capability. We defined a parameter grid for grid search for both algorithms. To enhance algorithm accuracy, an ensemble learning technique was incorporated to select the best classification model. A held-out validation mechanism was utilized, and HD datasets were used to validate the proposed model.

The proposed model was evaluated using different evaluation metrics. According to experimental outcomes on the selected feature dataset (CHDP), the stacking technique selected meta classifier (Random Forest) and obtained the higher performance in terms of accuracy, 99.25%, and greater ROC cure. The proposed stacking HD model experimental outcomes presented that the model obtained higher results in terms of accuracy compared with existing models. Due to its excellent results, the proposed stacking HD model is recommended for HD detection in IoT healthcare systems. In the future, we will incorporate deep learning, transfer learning, and federated learning techniques to design a more advanced system for the diagnosis of heart disease in the IOT healthcare system.

Data availability statement

The data sets utilized in this work were collected from a public repository. The Cleveland Heart Disease dataset is available at the link: https://www.kaggle.com/datasets/aavigan/cleveland-clinic-heart-disease-dataset. Heart Statlog Cleveland Hungary data set is available at the link: https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive. All methods were performed in accordance with the relevant guidelines and regulations.

Ethics statement

Author contributions

WJ: Funding acquisition, Resources, Validation, Visualization, Writing – original draft, Writing – review & editing. JL: Funding acquisition, Resources, Validation, Visualization, Writing – original draft, Writing – review & editing, Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Software, Supervision. AH: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. SK: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. RA: Writing – review & editing, Formal analysis, Visualization, Software. SA: Writing – review & editing, Methodology, Validation, Visualization, Software. MH: Data curation, Validation, Visualization, Writing – original draft, Writing – review & editing.

Funding

Acknowledgments

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Mendis S. *Global Atlas on Cardiovascular Disease Prevention and Control.* Geneva: World Health Organization (WHO) (2011).

2. Haq AU. A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms. *Mob Inf Syst.* (2018) 2018: 3860146. doi: 10.1155/2018/3860146

3. Durairaj M. A comparison of the perceptive approaches for pre-processing the data set for predicting fertility success rate. *Int J Control Theory Appl.* (2016) 9:255–60.

4. Kumar PM, Gandhi UD. A novel three-tier Internet of Things architecture with machine learning algorithm for early detection of heart diseases. *Comput Electr Eng.* (2018) 65:222–35. doi: 10.1016/j.compeleceng.2017.09.001

5. Mpanya D. Predicting mortality and hospitalization in heart failure using machine learning: a systematic literature review. *IJC Heart Vasc.* (2021) 34:100773. doi: 10.1016/j.ijcha.2021.100773

6. Haq AU. IIMFCBM: intelligent integrated model for feature extraction and classification of brain tumors using MRI clinical imaging data in IoT-healthcare. *IEEE J Biomed Health Inform.* (2022) 26:5004–12. doi: 10.1109/JBHI.2022.3171663

7. Detrano R, Janosi A. International application of a new probability algorithm for the diagnosis of coronary artery disease. *Am J Cardiol.* (1989) 64:304–10. doi: 10.1016/0002-9149(89)90524-9

8. Kahramanli H, Allahverdi N. Design of a hybrid system for diabetes and heart diseases. *Expert Syst Appl.* (2008) 35:82–9. doi: 10.1016/j.eswa.2007.06.004

9. Palaniappan S. Intelligent heart disease prediction system using data mining techniques. In: *2008 IEEE/ACS International Conference on Computer Systems and Applications.* Doha: IEEE (2008), p. 108–15. doi: 10.1109/AICCSA.2008.4493524

10. Olaniyi EO. Heart diseases diagnosis using neural networks arbitration. *Int J Intell Syst Appl.* (2015) 7:72. doi: 10.5815/ijisa.2015.12.08

11. Samuel OW. An integrated decision support system based on ANN and Fuzzy_AHP for heart failure risk prediction. *Expert Syst Appl.* (2017) 68:163–72. doi: 10.1016/j.eswa.2016.10.020

12. Liu X, Wang. A hybrid classification system for heart disease diagnosis based on the RFRS method. *Comput Math Methods Med.* (2017) 2017:8272091. doi: 10.1155/2017/8272091

13. Mohan S, Thirumalai C. Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access.* (2019) 7:81542–54. doi: 10.1109/ACCESS.2019.2923707

14. Li JP, Haq AU. Heart disease identification method using machine learning classification in e-healthcare. *IEEE Access.* (2020) 8:107562–82. doi: 10.1109/ACCESS.2020.3001149

15. Tiwari A, Chugh A, Sharma A. Ensemble framework for cardiovascular disease prediction. *Comput Biol Med.* (2022) 146:105624. doi: 10.1016/j.compbiomed.2022.105624

16. Gudadhe M. Decision support system for heart disease based on support vector machine and artificial neural network. In: *2010 International Conference on Computer and Communication Technology (ICCCT).* Allahabad: IEEE (2010), p. 741–5. doi: 10.1109/ICCCT.2010.5640377

17. Das R, Turkoglu I. Effective diagnosis of heart disease through neural networks ensembles. *Expert Syst Appl.* (2009) 36:7675–80. doi: 10.1016/j.eswa.2008.09.013

18. Bhatt CM, Patel P, Ghetia T, Mazzeo PL. Effective heart disease prediction using machine learning techniques. *Algorithms*. (2023) 16:88. doi: 10.3390/a16020088

19. Ozcan M, Peker S. A classification and regression tree algorithm for heart disease modeling and prediction. *Healthc Anal*. (2023) 3:100130. doi: 10.1016/j.health.2022.100130

20. Shukur BS, Mijwil MM. Involving machine learning techniques in heart disease diagnosis: a performance analysis. *Int J Electr Comput Eng*. (2023) 13:2177. doi: 10.11591/ijece.v13i2

21. Breiman L. Random forests. *Mach Learn*. (2001) 45:5–32. doi: 10.1023/A:1010933404324

22. Hastie T. Multi-class AdaBoost. *Stat Interface*. (2009) 2:349–60. doi: 10.4310/SII.2009.v2.n3.a8

23. Jansi Rani S, Chandran KS. Smart wearable model for predicting heart disease using machine learning: wearable to predict heart risk. *J Ambient Intell Humaniz Comput*. (2022) 13:4321–32. doi: 10.1007/s12652-022-03823-y

24. Doppala BP. A hybrid machine learning approach to identify coronary diseases using feature selection mechanism on heart disease dataset. *Distrib and arallel Databases*. (2021) 41:1–20. doi: 10.1007/s10619-021-07329-y

25. Pires IM. Machine learning for the evaluation of the presence of heart disease. *Procedia Comput Sci*. (2020) 177:432–7. doi: 10.1016/j.procs.2020.10.058

26. Al Ahdal A. Monitoring cardiovascular problems in heart patients using machine learning. *J Healthc Eng*. (2023) 2023:9738123. doi: 10.1155/2023/9738123

27. Saboor A. A method for improving prediction of human heart disease using machine learning algorithms. *Mob Inf Syst*. (2022) 2022:1410169. doi: 10.1155/2022/1410169

28. Shah D, Patel S. Heart disease prediction using machine learning techniques. *SN Comput Sci*. (2020) 1:1–6. doi: 10.1007/s42979-020-00365-y

29. Kishor A, Jeberson W. Diagnosis of heart disease using internet of things and machine learning algorithms. In: *Proceedings of Second International Conference on Computing, Communications, and Cyber-Security: IC4S 2020*. Singapore: Springer (2021), p. 691–702. doi: 10.1007/978-981-16-0733-2_49

30. Ganesan M, Sivakumar N. IoT-based heart disease prediction and diagnosis model for healthcare using machine learning models. In: *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*. Pondicherry: IEEE (2019), p. 1–5. doi: 10.1109/ICSCAN.2019.8878850

31. Nancy AA, Ravindran D. IoT-cloud-based smart healthcare monitoring system for heart disease prediction via deep learning. *Electronics*. (2022) 11:2292. doi: 10.3390/electronics11152292

32. Ahamed J, Manan Koli A, Ahmad K, Alam Jamal M, Gupta BB. CDPS-IoT: cardiovascular disease prediction system based on IoT using machine learning. *Int J Interact Multimedia Artif Intellig*. (2022) 7:78–86. doi: 10.9781/ijimai.2021.09.002

33. Ahmed F. An Internet of Things (IoT) application for predicting the quantity of future heart attack patients. *Int J Comput Appl*. (2017) 164:36–40. doi: 10.5120/ijca2017913773