Translation of article 10.22146/jnteti.v14i2.17157

Multi-Algorithm-Based Ensemble Voting Classifier and SMOTE Method for Heart Disease Classification

Dede Kurniadi[1], Asri Indah Pertiwi[1], Asri Mulyani[1]

[1] Program Studi Teknik Informatika, Jurusan Ilmu Komputer, Institut Teknologi Garut, Garut, Jawa Barat 44151, Indonesia

[Submitted: 19 November 2024, Revised: 23 January 2025, Accepted: 17 April 2025]
Corresponding Author: Dede Kurniadi (dede.kurniadi@itg.ac.id)

ABSTRACT — The heart is a vital organ responsible for pumping blood throughout the body. Hence, impairments can disrupt blood circulation and are the leading causes of mortality worldwide. World Health Organization (WHO) reported that, in 2021, the mortality rate attributed to heart disease reached a significant number. In Indonesia, the prevalence of heart disease attained 1.5%. Consequently, it is essential to prevent and detect heart disease at an early stage utilizing machine learning technologies. This study aims to develop a heart disease classification model using the naïve Bayes and random forest algorithms through the ensemble voting classifier approach. The data were obtained from Kaggle, comprising 1,000 records with 14 variables, including one classification target. Imbalanced data were handled using the synthetic minority oversampling technique (SMOTE), while feature selection was conducted in consultation with cardiologists to ensure clinical relevance. The model was trained using the naïve Bayes algorithm, random forest, and integration of both through the ensemble voting classifier method, in contrast to previous studies that only compared several algorithms to determine the highest accuracy. The test results showed that the model trained with the ensemble voting classifier yielded the best performance, with an accuracy, precision, recall, and F1 score of 98.28%, 98.41%, 98.41%, and 98.41%, respectively. This study demonstrates that the ensemble voting classifier method provides better accuracy than the individual algorithms. This model falls within the excellent classification category and is expected to contribute to the medical field and support the development of decision-support systems for diagnosing heart disease.

KEYWORDS — Ensemble Voting Classifier, Heart Disease Classification, Naïve Bayes, Random Forest.

I. INTRODUCTION

Health challenges in Indonesia are becoming increasingly diverse, with various diseases attacking the population—from mild to potentially life-threatening conditions [1]. Among these, heart disease has become a major concern. As an organ responsible for pumping blood, the heart plays a vital role in distributing oxygen and nutrients throughout the body. Any dysfunction in the heart can inhibit blood flow throughout the body [2]. Heart disease remains a global health issue, with its severity increasing each year, making this disease the leading cause of mortality worldwide [3].

The World Health Organization (WHO) reported that 17.9 million deaths worldwide in 2021 were attributed to cardiovascular disease, including heart disease [4]. According to the *Laporan Nasional Riskesdas 2018* (2018 National Basic Health Research) by the Ministry of Health of the Republic of Indonesia, the prevalence of heart disease diagnosed by cardiologists in Indonesia attained 1.5% in all age groups [5]. Therefore, optimal early prevention, detection, and management efforts are essential. Early detection plays a critical role in mitigating the risks associated with heart disease, thereby increasing individuals' chances to live healthier and longer [6]. It is particularly recommended for individuals over the age of 40 and those at risk, such as individuals with hypertension and diabetes [7].

Technological advancement makes it possible to address this issue using a system that supports medical personnel in the prevention and early detection of heart disease, utilizing machine learning to develop models capable of identifying heart disease. The ability of machine learning to analyze data quickly and accurately is expected to facilitate time and cost saving in the diagnosis process while reducing the risk of human errors [8]. Thus, the use of this machine learning technology can enhance the effectiveness of early detection and treatment of heart disease, reduce costs, and improve people's quality of life.

II. RELATED WORKS

Several prior studies have explored the use of machine learning technology in detecting and analyzing heart disease. These studies have compared various algorithms, such as naïve Bayes and random forest, to improve the accuracy and reliability of predictions. One study sought an algorithm that fitted the data used in heart disease classification with the decision tree algorithm, naïve Bayes, and random forest classifier [9]. This study utilized the Heart Attack Analysis & Prediction dataset from Kaggle, consisting of 918 rows of data and 12 attributes, with 11 attributes as input (age, sex, chest pain type, resting bp, cholesterol, fasting bs, resting ecg, max hr, exercise angina, oldpeak, st slope) and 1 attribute as output (heart disease). In this study, feature selection, outlier handling, and label handling were carried out in the data preprocessing stage. For testing, the dataset was divided into two parts, with a ratio of 80:20; namely, 80% of the data were used as training data and 20% as test data. In the test results of all models, the random forest classifier emerged as the most superior, with a score of 0.868 from grid search hyperparameter tuning and 0.852 from random search.

Reference [10] determined the best model for analyzing heart disease by applying various algorithms. By using several methods, this study aimed to enhance model performance to produce more accurate predictions and support a more precise diagnosis process. This study employed several algorithms: random forest, C45 Algorithm, logistic regression, and support

vector machine (SVM). The data used consisted of 300,000 data, with 18 variables and 1 target. In addition, classification performance was compared using the synthetic minority oversampling technique (SMOTE) and adaptive synthetic (ADASYN) methods. The feature extraction method was also utilized to identify the most influential variables. The results indicated that the random forest was the best algorithm, achieving an initial accuracy of 90.71%, which increased to 94.54% after the application of the oversampling technique.

Furthermore, another study compared decision trees, naïve Bayes, and random forest algorithm models for classifying heart disease [11]. A total of 319,795 data experienced data imbalance, which was handled using random undersampling techniques. Consequently, a dataset consisting of 54,746 data was obtained. The processed dataset was then divided into two parts for model training and testing, with a ratio of 80% (43,796 training data) and 20% (10,950 test data). The study results showed that the random forest algorithm achieved the best accuracy, with an accuracy value of 75%, a precision of 77%, a recall of 74%, and an F1 score of 76%.

Reference [12] compared seven machine learning classification methods: naïve Bayes, k-nearest neighbor (KNN), random forest, logistic regression, SVM, decision tree, and adaptive boosting (AdaBoost). The data used were 297 clean data from 303 collected records (6 data have incomplete variables) with 14 variables. This dataset was Cleveland Clinic Foundation data obtained from the UCI Machine Learning Repository. The experimental results indicated that the naïve Bayes algorithm provided the best performance, with an accuracy rate of 84.67%.

Another study involved a comparative approach by evaluating the naïve Bayes, random forest, and KNN algorithms for heart disease classification [13]. The dataset consisted of 304 data entries separated into 294 data for the training process and 10 remaining data used for testing. This data source was obtained from the Cleveland Clinic Foundation and adopted by the Hungarian Institute of Cardiology in Budapest. The results suggested that the naïve Bayes algorithm provided the highest accuracy, namely area under the curve (AUC) 0.91, calibration (CA) 0.84, F1 score 0.84, precision 0.839, and recall 0.84.

Based on these studies, the naïve Bayes and random forest algorithms have demonstrated superior performance in classifying heart disease. Both algorithms have shown consistent performance and offered high accuracy in detecting heart disease. However, previous studies only compared several algorithms to obtain the algorithm with the highest accuracy. Therefore, this study proposes a heart disease classification model that integrates two algorithms (naïve Bayes and random forest) and leverages the strengths of each algorithm using the ensemble voting classifier approach.

A. HEART DISEASE

Heart disease refers to a broad spectrum of medical conditions that involve various heart problems. These problems can occur in the heart's blood vessels, valves, or even the heart muscle. In addition, heart disease may result from other factors, such as infections and congenital abnormalities [14]. Due to various disorders that may occur, heart disease is considered one of the complex health problems and frequently necessitates proper medical treatment.

Numerous factors contribute to the severity of heart disease in patients, thereby necessitating medical treatment. One of the primary factors is the presence of cardiovascular risk factors and comorbid conditions, which significantly influence the risk of heart disease. Cardiovascular risk factors can be classified into two main groups: unmodifiable and modifiable. Unmodifiable risk factors include age, gender, and genetic history, all of which affect a person's likelihood of developing heart disease. Meanwhile, modifiable risk factors can be changed or managed through certain actions, including high blood pressure (hypertension), high cholesterol levels in the blood, smoking habits, diabetes, and being overweight or obese [2].

B. MACHINE LEARNING

Machine learning is a subdiscipline in artificial intelligence that allows systems to learn patterns from data naturally and automatically and improve their abilities through experiential learning, without requiring direct programming [15]. The focus of machine learning is to develop computer programs capable of accessing data and deriving knowledge from that data.

C. CLASSIFICATION

Classification is an essential process in data analysis, which aims at identifying patterns in the dataset and splitting data into different classes. This process involves building a model or function that can learn these patterns from data that has been given a class label. Using various algorithms and techniques, classification allows to predict the class of new data that does not yet have a class label. Classification models can be used in various fields, such as pattern recognition, image analysis, and biomedicine. By understanding the patterns in the data and classifying them accurately, superior decisions can be made, and valuable insights can be obtained from the data [16].

D. NAÏVE BAYES

Naïve Bayes is a classification approach based on Bayes' Theorem, a principle that utilizes the concepts of probability and statistics discovered by the British scientist, Thomas Bayes. Bayes' Theorem allows for estimates of the probability of future events based on prior knowledge [17].

The naïve Bayes algorithm, as one of the algorithms of machine learning, is widely employed for solving classification problems, especially in the context of text classification involving high-dimensional training data. Naïve Bayes estimates the probability of the target class based on the observed features, assuming that each feature is independent of each other, yet it can yield a fairly accurate classification [18].

According to [19], a formula that can be utilized in naïve Bayes is presented in (1).

$$P\,(D) = \frac{P\,(Ci) \times P\,(Ci)}{P\,(D)} \quad (1)$$

where $P\,(Ci|D)$ is the conditional probability of the category against the document, $P\,(D|Ci)$ is the conditional probability of the document against the category, $P(Cj)$ is the probability of the category or text to be classified, and $P(D)$ is the probability of the document or data.

E. RANDOM FOREST

Random forest is a machine learning method grounded in supervised machine learning that repeatedly applies the concept of decision trees, resulting in a collection of decision trees called forests [20]. It is an evolution of the classification and regression trees (CART) algorithm, which is closely related to the decision tree technique. The distinction between random forest and CART lies in its utilization of the bootstrap

aggregating (bagging) method and random feature selection, often referred to as random feature selection [21]. In its implementation, random forest constructs a large number of decision trees in parallel by employing the bagging technique, where each decision tree is constructed utilizing data samples randomly selected with replacement from the available dataset. Moreover, only a small subset of the features is considered in decision-making for each decision tree and these features are selected randomly. Equation (2) is used in the implementation of random forest.

$$Gini = 1 - \sum_{i=1}^{c} (pi)^2 \quad (2)$$

where pi is relative frequency and c is the number of classes. Equation (2) is a Gini equation that is used to determine decisions regarding splitting nodes in a decision tree branch.

F. ENSEMBLE VOTING CLASSIFIER

An ensemble voting classifier is a machine-learning approach that integrates multiple learning models to enhance prediction performance. This concept employs multiple learning algorithms to build several independent models, which are subsequently combined to produce the final prediction [18]. In the ensemble voting classifier process, each model votes or contributes according to its predictions. The votes from each model are then summed or averaged, and the class with the highest number of votes is designated as the final prediction.

III. METHODOLOGY

The method used in this study was the machine learning lifecycle (MLLC). MLLC refers to a series of steps or processes starting from data collection until the resulting model is ready to use [22]. This MLLC takes place progressively, moving forward, and can be repeated or iterative. Each iteration aims to improve the accuracy and performance of the model being developed. This study built a machine learning model using the Python programming language on the Jupyter Notebook tool version 6.4.12 in the Anaconda application. The stages carried out include data acquisition, data preprocessing, and model training and evaluation.

A. DATA ACQUISITION

Data acquisition was done through two stages. The initial stage involved data collection, during which raw data related to heart disease were searched and collected. The second stage involved data analysis, which was carried out to understand the characteristics of the data to be processed and used in machine learning classification modeling.

B. DATA PREPROCESSING

This stage is essential to ensure the quality and reliability of the data used to train the model. There were three activities carried out at this stage. The first activity was data balancing, which was conducted to handle data imbalance. The technique used for this data balancing was SMOTE. This technique ensures a more even distribution of data between the minority and majority classes [23]. The second activity involved feature engineering. At this stage, interviews were conducted with cardiologists to ascertain the attributes applicable for modeling. The final activity was data splitting, which was done to divide the dataset into different subsets for model training and testing.

C. MODEL TRAINING AND EVALUATION

At this stage, a heart disease classification model was constructed. This study applied a combination of naïve Bayes and random forest algorithms through the ensemble voting classifier method. Therefore, there were several activities performed, starting with building a naïve Bayes model. In implementing the model, data that had been previously prepared and selected were utilized. Equation (1) is used to implement naïve Bayes [19]. Furthermore, a random forest model was built. The Gini index formula was applied to determine decisions regarding splitting nodes in the decision tree branch. Equation (2) is the Gini index formula used in the implementation of random forest. Then, the ensemble voting classifier was implemented. Each classification model aims to achieve optimal performance while balancing the impact of individual weaknesses in various parts of the dataset. The ensemble voting classifier technique leverages the strengths of each model, thereby producing more accurate and stable predictions [24]. Various ensemble methods have been introduced as they have demonstrated superior performance on different heart disease datasets [25]. Finally, all models that had been created and built were then evaluated. At this evaluation stage, an evaluation model (that includes accuracy, precision, recall, and F1 score) was used.

IV. RESULTS

A. DATA ACQUISITION

At the data acquisition stage, two activities were carried out: data collection and data analysis. The following is a detailed explanation of each stage.

1) DATA COLLECTION

In this stage, a dataset on heart disease datasets was collected online from one of the dataset collection websites, namely Kaggle. The data collection employed in this study can be seen at kaggle.com/datasets/jocelyndumlao/cardiovascular-disease-dataset [26]. This dataset consisted of 1,000 data with 14 variables. A total of 13 variables were independent variables and 1 variable was a dependent variable or target. The target variable was a binary target, while the other 13 variables were categorical and numeric data types, as presented in Table I.

2) DATA ANALYSIS

At this stage, data analysis was conducted to determine the characteristics of the previously collected dataset. There were several activities carried out to analyze data. First, a missing value analysis was conducted to determine the number of empty or missing values in each variable in the dataset. The dataset was stored in a data variable and analyzed using functions from the Pandas library. The df.isnull().sum() function was used to analyze missing values in the data. The result of the missing value analysis was the collected dataset that was free from missing values; hence, it could be used for further analysis.

Subsequently, data duplication analysis was performed. The df.duplicated().sum() function was employed to analyze data duplication in the dataset. The results of the data duplication analysis are illustrated in Figure 1. It can be seen from the figure that the dataset is devoid of duplicates in all its variables. Therefore, the dataset can be further analyzed through the outlier data stage. An outlier analysis was conducted to identify and address data significantly different from the majority. It was performed by utilizing the Seaborn library with the df[columns_list].boxplot(figsize=(1 2, 6)) function. Outlier information was visualized using the boxplot of the program's processing results, as depicted in Figure 2.

As shown in Figure 2, the data distribution does not exhibit any outliers. As a result, further analysis, namely class

TABLE I
VARIABELS IN THE DATASET

No.	Variabel	Tipe Data
1	patientid	Numerik
2	age	Numerik
3	gender	Kategorial
4	restingbp	Numerik
5	serumcholestrol	Numerik
6	fastingbloodsugar	Kategorial
7	chestpain	Kategorial
8	restingelectro	Kategorial
9	maxheartrate	Numerik
10	exerciseangina	Kategorial
11	oldpeak	Numerik
12	slope	Kategorial
13	noofmajorvessels	Kategorial
14	clasification (target)	Kategorial

```
Jumlah baris duplikat: 0

Baris duplikat:
 Empty DataFrame
Columns: [patientid, age, gender, chestpain, restingBP, serumchole
strol, fastingbloodsugar, restingrelectro, maxheartrate, exercisea
ngia, oldpeak, slope, noofmajorvessels, target]
Index: []
```

Figure 1. Results of data duplication.

distribution, was carried out. The class distribution analysis on the heart disease dataset label was performed to assess the proportion of each class. This dataset comprised two binary classes: class 0 and class 1. Class 0 indicates the absence of heart disease, while class 1 indicates the presence of heart disease. The number of classes in this dataset was distributed using the df['target'].value_counts().tolist() function. The distribution of data classes in the dataset is shown in Table II. As shown in this table, the amount of data in both classes is different. Class 0 (representing the absence of heart disease) contains 420 data, while class 1 (representing the presence of heart disease) contains 580 data. This discrepancy indicates that the data is imbalanced.

Several analyses conducted on the heart disease dataset indicated one problem, namely data imbalance. Imbalance in data can significantly impact the classification results [27]. Therefore, this problem must be resolved to prevent it from affecting the performance of the model created. The resolution of this data imbalance problem is carried out in the next stage, namely the data preprocessing stage.

B. DATA PREPROCESSING

There were three activities carried out in the data preprocessing stage. The following are details of each activity.

1) DATA BALANCING (SMOTE)

Table II shows the presence data imbalance, so it is necessary to carry out a data balancing process by applying the SMOTE technique. This SMOTE technique performs over-sampling by creating synthetic samples for the minority class, based on analysis and interpolation of existing data to improve data distribution [23]. However, if the data are not too complex, the SMOTE technique can introduce synthetic bias that is not present in the original data. This shortcoming can affect the model's performance on real data, especially if the model learns more from synthetic patterns than from original patterns. The initial step in implementing SMOTE was to visualize the

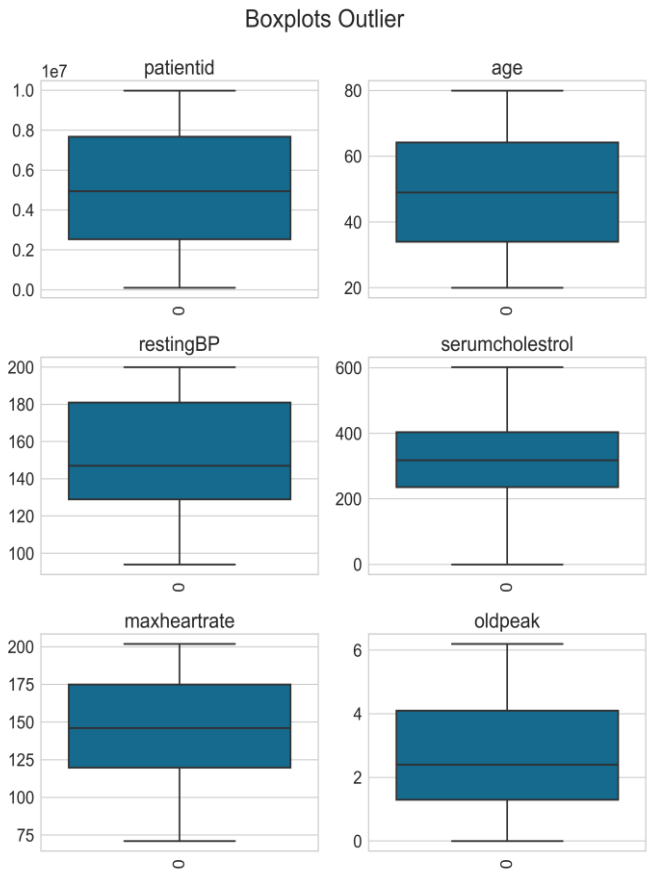

Figure 2. Outlier information.

TABLE II
DATA CLASS DISTRIBUTION

No.	Class	Number of Data
1	0	420
2	1	580

distribution of the target class using the countplot function from the Seaborn library. This visualizat ion helps in understanding the proportion of data in the minority and majority classes. Furthermore, the input or feature attributes (x) and output or target (y) were separated from the dataset to facilitate the data balancing process. The feature attribute contained medical record information, such as restingbp and serumcholestrol, while the target indicated the presence or absence of heart disease in the patient. Subsequently, the SMOTE object was initialized with a random seed for reproducibility using the default parameter k_neighbors = 5. The SMOTE(random_state = 99) function was then applied to perform linear interpolation in creating synthetic samples of the minority class. In validating the synthetic data that had been created, the model was retrained with a balanced dataset and tested to ensure that the synthetic data did not negatively affect the model performance. After SMOTE was applied, a balanced class distribution was generated with the X_resampled function, y_resampled = smote.fit_resample(X, y), where X is a feature attribute or input and Y is the target or output. The results of the program processing before and after the application of SMOTE are presented in Figure 3.

Figure 3 shows that the data imbalance has been resolved using the SMOTE technique. After the data were balanced using SMOTE, the data in class 0 (representing the absence of

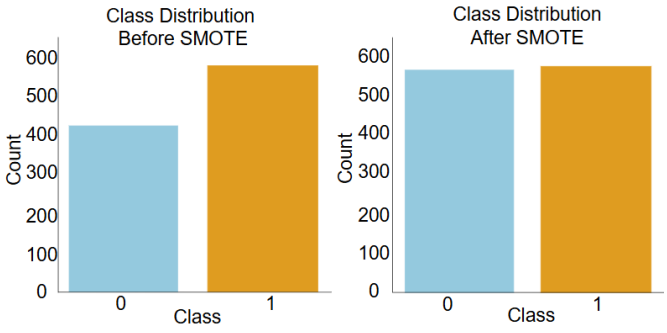

Figure 3. Results of SMOTE technique implementation.

heart disease), which was originally 420 data, increased by 160 data, to 580. Meanwhile, the data in class 1 (representing the presence of heart disease), remained at 580 data. Thus, the total amount of data after applying SMOTE was 1,160 data.

2) FEATURE ENGINEERING

In the feature engineering process, an expert interview was conducted with a cardiologist, dr. Gusti Made Odi Sidharta, Sp.JP. Based on the interview, the variables in the dataset could be applied and utilized in classifying heart disease. However, among the 14 variables in the dataset, one variable—namely the "patiented"—was excluded from further process as it did not affect the classification of heart disease. Meanwhile, the remaining 13 variables were still used as dr. Gusti Made Odi Sidharta, Sp. JP emphasized their importance in determining whether someone has heart disease or not since heart disease is very dangerous. Table III presents the variables obtained from selection results that can be used in the next process. Table III is the result of the feature engineering process carried out manually through interviews with cardiologists. The next process used the remaining 13 variables that were obtained at this stage.

3) DATA SPLITTING

The last activity in the data preprocessing stage was data splitting. This activity was carried out to divide the data into two subsets: training and testing. The division of the subsets used a ratio of 90:10, with 90% of the data (1044 data) as training data and 10% (160 data) as test data. The results of the data splitting were then used to train and test the naïve Bayes, random forest, and ensemble voting classifier models.

C. MODEL TRAINING AND EVALUATION

At this stage, several activities were done to build and evaluate the model for heart disease classification.

1) NAÏVE BAYES MODEL

The implementation or construction of the naïve Bayes model began with the initialization of the naïve Bayes model using GaussianNB from the scikit-learn library. This algorithm was chosen since this model did not require many hyperparameter configurations and only used simple default settings. Then, the model was trained using training data of 90% of the data from the dataset. The function used in training this model was nb_model.fit(X_train, y_train).

At this stage, the model was trained to learn the data pattern. After the model was trained, the prediction probability was calculated using nb_model.predict_proba(X_test)[:, 1] and the target value prediction was made on the test data using the nb_model.predict(X_test) function. The calculation of prediction probability was done utilizing (1). The results of the

TABLE III
RESULTS OF FEATURING ENGINEERING PROCESS

No.	Variable	Data Type
1	age	Int64
2	gender	Int64
3	restingbp	Int64
4	serumcholestrol	Int64
5	fastingbloodsugar	Int64
6	chestpain	Int64
7	restingelectro	Int64
8	maxheartrate	Int64
9	exerciseangina	Int64
10	oldpeak	float64
11	slope	Int64
12	noofmajorvessels	Int64
13	target	Int64

prediction probability calculation were subsequently used in the model evaluation to generate a confusion matrix evaluation (accuracy, precision, recall, and F1 score). The results of processing the naïve Bayes classification model implementation program are shown in Figure 4.

As illustrated in Figure 4, the evaluation results of heart disease classification modeling using the naïve Bayes algorithm demonstrated varying performance in several evaluation metrics. This algorithm achieved an accuracy level of 67.24%, indicating that over half of the predictions generated by the model matched the actual labels. In addition, the naïve Bayes algorithm also produced a fairly high precision value, with 90.32%, indicating that of all the positive predictions generated, most of them were actual cases of heart disease. However, the recall was considerably low, at 44.44%, suggesting that many positive cases were not detected by the model. This resulted in the F1 score value, which is the average between precision and recall, to be at 59.57%. Further evaluation is also presented through the confusion matrix, as shown in Figure 5(a) and Figure 5(b), which provides a more detailed picture of the number of correct and incorrect predictions generated by the model for each category.

Figure 5 was generated through program processing. Figure 5(a) displays the model performance analysis by providing information about the number of correct and incorrect predictions generated by the model compared to the actual values of the data. In the training data, 478 data in class 0 were predicted as true negative (TN) and 272 data in class 1 were predicted as true positive (TP). However, 49 data were detected as false positive (FP) and 245 as false negative (FN). Meanwhile, Figure 5(b) shows the confusion matrix of the model prediction results using the test data. The algorithm successfully identified a large amount of data correctly. Among all the data tested, 50 data were accurately predicted as TN, specifically data from class 0 and were correctly predicted by the model. In addition, 28 data were categorized as TP, indicating that the data came from class 1 and were also accurately predicted by the model. Nonetheless, not all predictions generated by the model corresponded with the actual category. Three data were predicted as FP. These data originated from class 0, but they were predicted as class 1 by the model. In addition, 35 data were predicted as FN, which were data that should be classified as class 1, but were predicted as class 0 by the model. This confusion matrix provides a more detailed picture of the model's performance, both in identifying

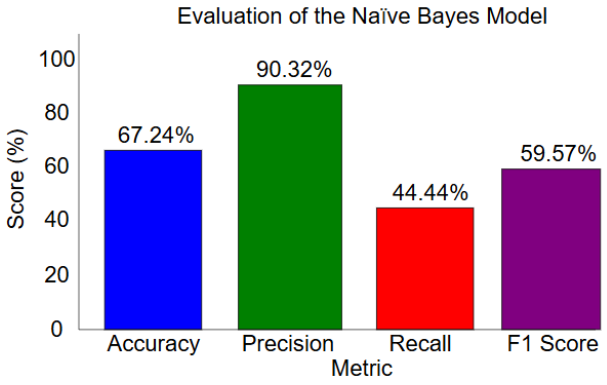

Figure 4. Results of naïve Bayes model evaluation.

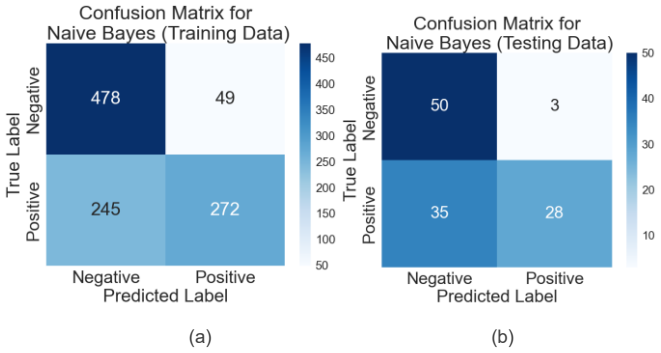

Figure 5. Confusion matrix of naïve Bayes, (a) training data and (b) test data.

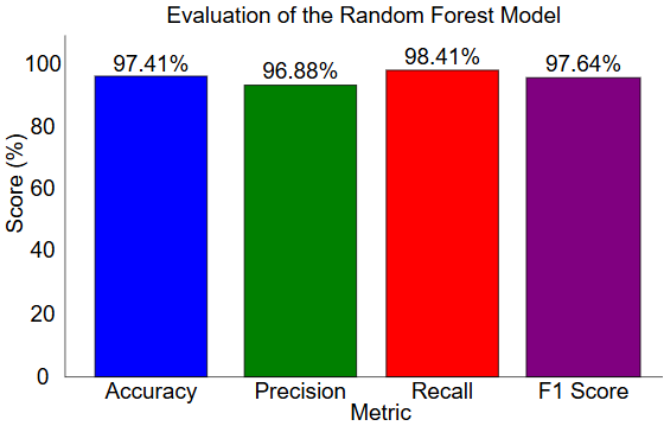

Figure 6. Evaluation results of the random forest model.

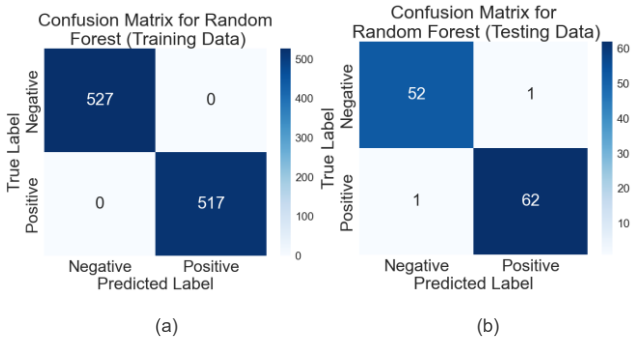

Figure 7. Confusion matrix of the random forest, (a) training data, (b) test data.

the correct class and in making classification errors, which can be used as a reference for further model improvement.

2) RANDOM FOREST MODEL

The subsequent stage involved the implementation or construction of a random forest model. Heart disease classification modeling with the random forest algorithm commenced with the initialization of random forest, employing 90% of the dataset as training data, through the RandomForestClassifier(criterion='gini', random_state=77) function. This model was created using the Gini splitting criteria—used to determine the best split at each node—and using random state parameters to ensure consistent results. Then, the model was trained using the fit(X_train, y_train) function, allowing it to learn patterns from the data. Moreover, the prediction probability was calculated. In the modeling, the Gini index formula in (2) was used and predictions were made using predict(X_test) and test data as much as 10% of the dataset. Subsequently, it was continued with the model evaluation stage. The results of the program processing from the random forest model evaluation are shown in Figure 6.

Figure 6 shows the percentage of the random forest model evaluation results built for heart disease classification. This model achieved excellent performance, with an accuracy of 97.41%, a precision of 96.88%, a recall of 98.41%, and an F1 score of 97.64%. These evaluation results clearly show the superiority of the random forest model over the previous model, with significant improvements in each evaluation metric used. This high percentage indicates that the random forest can classify data with a better level of accuracy and suitability.

In addition, further evaluation was performed using the confusion matrix, which is shown in Figure 7(a) and Figure 7(b). Both figures provide a more detailed visualization of the model's prediction distribution, including the model's success rate in identifying the correct category and avoiding misclassification.

Figure 7 was generated through program processing. In Figure 7(a), the results of the random forest model evaluation visualized through the confusion matrix provide a clear depiction of the model's performance on the training data. A total of 527 data were accurately classified as TN, demonstrating that the model was able to identify negative cases of heart disease with high accuracy. In addition, 517 data were also successfully classified correctly as TP, suggesting the model's ability to recognize data with heart disease. In addition, the results showed that no wrong predictions were identified in this training data, whether as a FP or FN. These results indicate that the random forest model can accurately map the training data. Meanwhile, in Figure 7(b), the confusion matrix from the evaluation results using test data shows several prediction errors. One datum was incorrectly classified as class 1, where it should be in class 0 (FP). In addition, one datum that was included in class 1 was predicted as class 0 (FN). Nevertheless, the model successfully identified 52 data as TN and 62 data as TP. Despite some errors, the random forest model exhibits quite good performance on the test data, but less accurate than its performance on the training data.

3) ENSEMBLE VOTING CLASSIFIER

The final stage involved the implementation of the ensemble voting classifier that combined the naïve Bayes and random forest algorithms. This implementation commenced with the initialization of the naïve Bayes and random forest models, followed by the initialization of the ensemble voting classifier on both models using the VotingClassifier function. Given that only two algorithms were combined, the soft voting with the voting function = 'soft'the was used as the ensemble. The model with this ensemble method allows the final decision

to be made based on the average probability generated from the naïve Bayes and random forest algorithms. In this case, naïve Bayes produces prediction probabilities under the assumption of independence between features. The approach used in naïve Bayes can simplify the process of calculating the joint probability of features to predict the target class. In contrast, random forest produces prediction probabilities by building several decision trees and combining the prediction results from each tree. By integrating the two models using this ensemble voting classifier, the error distribution that may occur in individual models can be minimized. After ensemble initialization, the process continued by training the ensemble model using training data (90% of the dataset), with the fit() function. Then, the prediction probability was calculated and predictions using test data (10% of the dataset) was made. The process ended with model evaluation. The results of processing the evaluation program on this model are shown in Figure 8.

As seen in Figure 8, the classification model employing the ensemble voting classifier method yielded very satisfactory evaluation results. This model attained an accuracy of 98.28% and precision value of 98.41%, signifying that most of the positive predictions were correct. In addition, the model also demonstrated stable performance in correctly identifying data, as evidenced by the recall value of 98.41%. This ensured that the majority of patients with heart disease were correctly identified. Overall, this model produced an F1 score value of 98.41%, indicating a balance between precision and recall. The results of this evaluation emphasize that the integration of the naïve Bayes algorithm and random forest classifier using the ensemble voting classifier is suitable and can strengthen the reliability of the model in predicting heart disease. If the two algorithms do not match, the resulting accuracy is likely to decrease compared to the utilization of a single algorithm.

In addition, to understand more about the model performance, further analysis was conducted using a confusion matrix. Figure 9(a) and Figure 9(b), which were generated through program processing, are visualizations of the confusion matrix. As illustrated in Figure 9(a), the confusion matrix with training data yielded no errors, with 527 correctly predicted in class 0 and 517 correctly predicted in class 1. However, as shown in Figure 9(b), the application of the ensemble voting classifier method produced a small error in the use of test data. Specifically, only one datum that was misclassified as FP and another as FN. Aside from that, there were no more errors. Meanwhile, 52 remaining data were correctly predicted to be in class 0 and 62 data were in class 1.

Based on the overall test results of the model consisting of the naïve Bayes algorithm, random forest, and ensemble voting classifier, the evaluation results indicate that the best performance is attained by a model that integrated several algorithms, namely the ensemble voting classifier. Each model underwent careful testing, and its performance results are presented in detail in Table IV. This table describes the comparison of various evaluation metrics, such as accuracy, precision, recall, and F1 score.

As presented in Table IV, the ensemble voting classifier model outperformed naïve Bayes and random forest modeling individually. Although the recall percentage remained the same across algorithms, the ensemble voting classifier model demonstrated superiority in other metrics. It attained an accuracy of 98.28%, precision of 98.41%, and F1 score of 98.41%. These outcomes confirm that integrating multiple

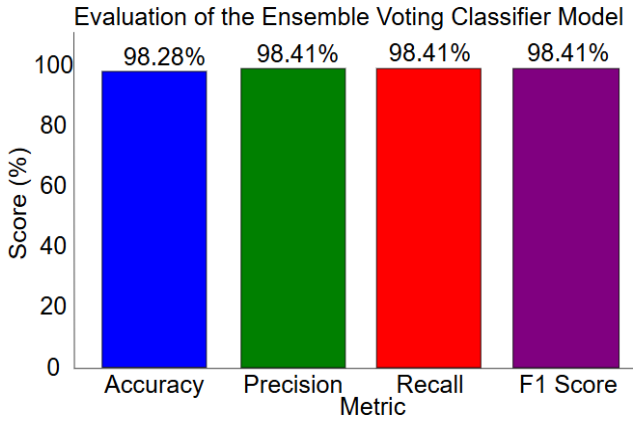

Figure 8. Evaluation results of the ensemble voting classifier model.

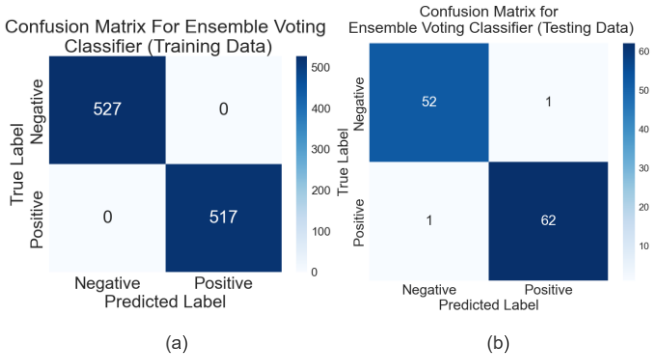

Figure 9. Confusion matrix of the ensemble voting classifier, (a) training data and (b) test data.

algorithms in one model can improve the accuracy and reliability of predictions.

V. DISCUSSION

This study successfully constructed a classification model for heart disease by integrating the naïve Bayes and the random forest algorithm through an ensemble voting classifier. The ensemble voting classifier was used to combine both algorithms, leveraging the strengths of each algorithm used. With a processed heart disease dataset of 1,160 data, the implementation of the ensemble voting classifier was successfully applied and provided accurate, consistent performance. In addition, it succeeded in increasing accuracy compared to using the naïve Bayes algorithm alone.

The results of this study have proven that the application of an ensemble voting classifier could be used to create a more effective and accurate heart disease classification model. This is in line with previous studies that have examined the performance of heart disease classification models by comparing the naïve Bayes and random forest algorithms [9]–[13]. The results of each related study are shown in Table V.

Based on Table V, this study not only focuses on the comparison between two algorithms, namely naïve Bayes and random forest, but also provides additional contributions by implementing the ensemble voting classifier method. This method was applied with the aim of enhancing the accuracy and stability of the heart disease classification model. The results of this study indicate that the utilization of ensemble voting classifiers can produce higher accuracy when compared to the individual performance of the naïve Bayes or random forest algorithm alone. By effectively combining the strengths of both algorithms, the ensemble method can enhance the overall

TABLE IV
EVALUATION RESULTS OF ALL THE MODELS

Classification Model	Evaluation Metrics (%)			
	Accuracy	*Precision*	*Recall*	*F1 Score*
Naïve Bayes	67.24	90.32	44.44	59.57
Random forest	97.41	96.88	98.41	97.64
Ensemble voting classifier	98.28	98.41	98.41	98.41

TABLE V
COMPARISON RESULTS OF ACCURACY RESULTS

Research	Method	Accuracy Results (%)
This research	Ensemble voting classifier, algoritma naïve Bayes dan random forest	Naïve Bayes: 67.24 Random forest: 97.41 Ensemble voting classifier: 98.28
[9]	Decision tree, naïve Bayes, dan random forest classifier	Decision tree: 84.40 Naïve Bayes: 85.00 Random forest classifier: 85.20
[10]	Random forest classifier, C45 algorithm, logistic regression, dan (SVM)	Random forest classifier: 94.54 C45 algorithm: 91.74 Logistic regression: 76.27 SVM: 76.24
[11]	Decision tree, naïve Bayes, dan random forest	Decision tree: 72.00 Naïve Bayes: 71.00 Random forest: 75.00
[12]	Naïve Bayes, KNN, random forest, logistic regression, SVM, decision tree, and AdaBoost	Naïve Bayes: 84.67 KNN: 73.00 Random forest:81.70 Logistic regression: 84.30 SVM: 81.00 Decision tree: 74.00 AdaBoost: 71.30
[13]	Naïve Bayes, random forest classifier, dan k-nearest neighbor (KNN)	Naïve Bayes: 91.00 Random forest: 89.90 KNN: 68.60

performance of the model and provide more accurate and reliable prediction results.

This heart disease classification modeling demonstrates that the application of the ensemble voting classifier method can contribute significantly to the medical field, especially in the diagnosis of heart disease. By combining naïve Bayes and random forest, this model enhances accuracy and speed of prediction, thereby facilitating faster and more precise clinical decision-making. The potential integration of these results into clinical decision support systems in healthcare facilities can efficiently enhance the identification of high-risk patients, enable more effective early detection, and contribute to improving the patients' quality of life.

VI. CONCLUSION

This study successfully built a heart disease classification model by implementing an ensemble voting classifier to combine the naïve Bayes and random forest algorithms. A problem was identified during the process, namely data class imbalance. The SMOTE technique was utilized to resolve this problem. It has proven to be effective in handling data class imbalance. The data splitting process involved dividing the data into train and test data, adhering to a 90:10 ratio. The evaluation results indicated an increase in performance following the implementation of the ensemble voting classifier. The implementation of an ensemble voting classifier to combine the naïve Bayes and random forest algorithms in the heart disease classification model has demonstrated very accurate and stable performance. This model attained an accuracy of 98.28%, a precision of 98.41%, a recall of 98.41%, and an F1 score of 98.41%. Despite one percentage similarity in the recall metric, the ensemble voting classifier method offers more comprehensive and superior results attributed to the contribution of the other three base classifiers (accuracy, precision, and F1 score). This indicates that combining multiple algorithms in one ensemble model can improve evaluation results and increase reliability in crucial medical prediction models, such as heart disease classification.

This study utilized an ensemble voting classifier to combine two algorithms and leveraged each strength (naïve Bayes dan random forest) to enhance the model's accuracy. Nevertheless, future studies are suggested to explore other ensemble algorithms and methods to improve the model's accuracy, thereby achieving a more accurate result.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

AUTHORS' CONTRIBUTIONS

Conceptualization, Dede Kurniadi and Asri Indah Pertiwi; methodology, Dede Kurniadi; software, Asri Indah Pertiwi; data and validation, Dede Kurniadi and Asri Indah Pertiwi; writing—original draft preparation, Dede Kurniadi and Asri Indah Pertiwi; writing—revieweing and editing, Dede Kurniadi, Asri Indah Pertiwi, and Asri Mulyani; visualization, Asri Indah Pertiwi; supervision, Dede Kurniadi and Asri Mulyani.

ACKNOWLEDGMENT

Gratitude is extended to the Institut Teknologi Garut for funding this research and to Dr. Gusti Made Odi Sidharta, Sp.JP. who has taken the time to be interviewed during the feature engineering process.

REFERENCES

[1] L.P.C. Dewi. "Jenis, gejala, dan penyebab penyakit jantung." Access date: 13-Mar-2024. [Online]. Available: https://rs-soewandhi.surabaya.go.id/jenis-gejala-dan-penyebab-penyakit-jantung/

[2] S.D. Sawu, A.A. Prayitno, and Y.I. Wibowo, "Analisis faktor risiko pada kejadian masuk rumah sakit penyakit jantung koroner di Rumah Sakit Husada Utama Surabaya," *J. Sains Kesehat.*, vol. 4, no. 1, pp. 10–18, Jul. 2022, doi: 10.25026/jsk.v4i1.856.

[3] M. Ardiana, *Buku Ajar Prevensi dan Rehabilitasi Jantung*. Surabaya, Indonesia: Airlangga University Press, 2022.

[4] World Health Organization. "Cardiovascular diseases." Access date: 7-Aug-2024. [Online]. Available: https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1

[5] Tim Riskesdas 2018, *Laporan Nasional Riskedas 2028*. Jakarta, Indonesia: Lembaga Penerbit Badan Penelitian dan Pengembangan Kesehatan, 2019.

[6] Y.P. Santosa. "Memahami pentingnya cek kesehatan jantung." Primaya Hospital. Access date: 8-Jul-2024. [Online]. Available: https://primayahospital.com/jantung/pentingnya-cek-kesehatan-jantung/

[7] Direktorat Jenderal Pencegahan dan Pengendalian Penyakit. "Pemeriksaan, Gejala, dan Diet untuk Jantung." Access date: 2-Jun-2024.

[Online]. https://p2p.kemkes.go.id/pemeriksaan-gejala-dan-diet-untuk-jantung/

[8] A.M.A. Rahim, I.Y.R. Pratiwi, and M.A. Fikri, "Klasifikasi penyakit jantung menggunakan metode synthetic minority over-sampling technique dan random forest clasifier," *Indones. J. Comput. Sci.*, vol. 12, no. 5, pp. 2995–3011, Oct. 2023, doi: 10.33022/ijcs.v12i5.3413.

[9] J.D. Muthohhar and A. Prihanto, "Analisis perbandingan algoritma klasifikasi untuk penyakit jantung," *J. Inform. Comput. Sci. (JINACS)*, vol. 04, no. 3, pp. 298–304, Mar. 2023, doi: 10.26740/jinacs.v4n03.p298-304.

[10] A.F.N. Masruriyah *et al.,* "Evaluasi algoritma pembelajaran terbimbing terhadap dataset penyakit jantung yang telah dilakukan oversampling," *MIND (Multimed. Artif. Intell. Netw. Database) J.*, vol. 8, no. 2, pp. 242–253, Dec. 2023, doi: 10.26760/mindjournal.v8i2.242-253.

[11] D.H. Depari, Y. Widiastiwi, and M.M. Santoni, "Perbandingan model decision tree, naive Bayes dan random forest untuk prediksi klasifikasi penyakit jantung," *Inform., J. Ilmu Komput.*, vol. 18, no. 3, pp. 239–248, Dec. 2022, doi: 10.52958/iftk.v18i3.4694.

[12] Ratnasari, A.J. Wahidin, A.E. Setiawan, and P. Bintoro, "Machine learning untuk klasifikasi penyakit jantung," *Aisyah J. Inform. Electr. Eng.*, vol. 6, no. 1, pp. 145–150, Feb. 2024, doi: 10.30604/jti.v6i1.272.

[13] A. Samosir, M.S. Hasibuan, W.E. Justino, and T. Hariyono, "Komparasi algoritma random forest, naïve Bayes dan k-nearest neighbor dalam klasifikasi data penyakit jantung," in *Pros. Semin. Nas. Darmajaya*, 2021, pp. 214–222.

[14] B. Asrun and I. Irmayani, "Penerapan konsep non-deterministic finite automata dalam diagnosa penyakit jantung," *Dewantara J. Technol.*, vol. 3, no. 1, pp. 122–125, May 2022, doi: 10.59563/djtech.v3i1.184.

[15] P.D. Kusuma, *Machine Learning Teori, Program, dan Studi Kasus*. Yogyakarta, Indonesia: Deepublish, 2020.

[16] P.B.N. Setio, D.R.S. Saputro, and B. Winarno, "Klasifikasi dengan pohon keputusan berbasis algoritme C4.5," in *PRISMA, Pros. Semin. Nas. Mat.*, 2020, pp. 64–71.

[17] Rayuwati, H. Gemasih, and I. Nizar, "Implementasi algoritma naive Bayes untuk memprediksi tingkat penyebaran Covid," *J. Ris. Rumpun Ilmu Tek.*., vol. 1, no. 1, pp. 38–46, Apr. 2022, doi: 10.55606/jurritek.v1i1.127.

[18] M. Al-Husaini, P.A. Saputra, M. Renaldi, and R.A. Maulana, *Prediksi Tsunami dengan Metode Ensemble Machine Learning*. Jambi, Indonesia: PT. Sonpedia Publishing Indonesia, 2024.

[19] Sarwido, G.W.N. Wibowo and M.A. Manan, "Penerapan algoritma naive Bayes untuk prediksi heregistrasi calon mahasiswa baru," *J. Tek. Inform.*, vol. 1, no. 1, pp. 1–10, Feb. 2022, doi: 10.02220/jtinfo.v1i1.126.

[20] S. Saadah and H. Salsabila, "Prediksi harga Bitcoin menggunakan metode random forest," *J. Komput. Terap.*, vol. 7, no. 1, pp. 24–32, Jun. 2021, doi: 10.35143/jkt.v7i1.4618.

[21] M.R. Adrian, M.P. Putra, M.H. Rafialdy, and N.A. Rakhmawati, "Perbandingan metode klasifikasi random forest dan SVM pada analisis sentimen PSBB," *J. Inform. UPGRIS*, vol. 7, no. 1, pp. 36–40, 2021, doi: 10.26877/jiu.v7i1.7099.

[22] I. Daqiqil, *Machine Learning: Teori, Studi Kasus dan Implementasi Menggunakan Python*, 1st ed. Riau, Indonesia: UR PRESS, 2021.

[23] A.J. Mohammed, M.M. Hassan, and D.H. Kadir, "Improving classification performance for a novel imbalanced medical dataset using SMOTE method," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 3, pp. 3161–3172, Jun. 2020, doi: 10.30534/ijatcse/2020/104932020.

[24] M. Ardiansyah, "Model ensemble algoritma naive Bayes dan random forest dalam klasifikasi penyakit paru-paru untuk meningkatkan akurasi," *SMARTLOCK, J. Sains dan Teknol.*., vol. 2, no. 2, pp. 32–38, Dec. 2023, doi: 10.37476/smartlock.v2i2.4407.

[25] S. Bashir *et al.*, "A knowledge-based clinical decision support system utilizing an intelligent ensemble voting scheme for improved cardiovascular disease prediction," *IEEE Access*, vol. 9, pp. 130805–130822, Sep. 2021, doi: 10.1109/ACCESS.2021.3110604.

[26] J. Dumlao. "Cardiovascular Disease Dataset." Kaggle. Access date: 28-Apr-2024. [Online]. Available: https://www.kaggle.com/datasets/jocelyndumlao/cardiovascular-disease-dataset

[27] F. Rahman and Mustikasari, "Optimization of student graduation predictions on time using binning and synthetic minority oversampling technique (SMOTE)," *Jagti*, vol. 4, no. 1, pp. 30–36, Feb. 2024, doi: 10.24252/jagti.v4i1.77.