The Severity Prediction of The Binary And Multi-Class Cardiovascular Disease - A Machine Learning-Based Fusion Approach

Hafsa Binte Kibria
Department of Electrical & Computer Engineering
Rajshahi University of Engineering & Technology,
Rajshahi-6204,Bangladesh
hafsabintekibria@gmail.com

Abdul Matin
Department of Electrical & Computer Engineering
Rajshahi University of Engineering & Technology,
Rajshahi-6204,Bangladesh
ammuaj.cseruet@gmail.com

March 10, 2022

Abstract

In today's world, a massive amount of data is available in almost every sector. This data has become an asset as we can use this enormous amount of data to find information. Mainly health care industry contains many data consisting of patient and disease-related information. By using the machine learning technique, we can look for hidden data patterns to predict various diseases. Recently CVDs, or cardiovascular disease, have become a leading cause of death around the world. The number of death due to CVDs is frightening. That is why many researchers are trying their best to design a predictive model that can save many lives using the data mining model. In this research, some fusion models have been constructed to diagnose CVDs along with its severity. Machine learning(ML) algorithms like artificial neural network, SVM, logistic regression, decision tree, random forest, and AdaBoost have been applied to the heart disease dataset to predict disease. Randomoversampler was implemented because of the class imbalance in multiclass classification. To improve the performance of classification, a weighted score fusion approach was taken. At first, the models were trained. After training, two algorithms' decision was combined using a weighted sum rule. A total of three fusion models have been developed from the six ML algorithms. The results were promising in the performance parameter. The proposed approach has been experimented with different test training ratios for binary and multiclass classification problems, and for both of them, the fusion models performed well. The highest accuracy for multiclass classification was found as 75%, and it was 95% for binary. The code can be found in : github/hafsakibria/Weightedscorefusion

motivation & introdu

Keywords Artificial neural network · random forest · decision tree · adaboost · support vector machine · logistic regression · cardiovascular disease · weighted score fusion

1 Introduction

Cardiovascular diseases (CVD) are caused by the heart and blood vessels [1]. It is one of the significant causes of death around the world. Annually, it kills more people than any other cause. In 2016, the death rate was around 17.9 million, which represents 31% of worldwide mortality. Among them, 85% are because of stroke and heart attack. Most cardiovascular disorders can be avoided by controlling some approaches like unhealthy diet, imbalanced lifestyle, physical inactivity, and excessive alcohol usage. According to the estimation of WHO, almost 23.6 million deaths will occur because of CVDs, mainly from stroke and heart disease, by 2030. This estimation shows that cardiovascular disease remains the single major cause of death until now. So people with a high risk of cardiovascular diseases need to be detected. That is why diagnosing the disease correctly is necessary and proper treatment can also reduce the risk of further severity of the disease. A clear understanding of the risk is required to improve the diagnosis. In the traditional approach, diseases are identified by analyzing patients' symptoms and medical history, for example, ECG tests, blood sugar levels, blood pressure, cholesterol. This process is time-consuming and expensive. With the aid of machine

→ death statistics for major dis

learning, it becomes simple. This process saves much time and therefore improves the efficiency of the diagnosis. This is one of the reasons we have developed some fusion models to classify heart disease efficiently. Here heart disease has been classified using some machine learning-based fusion models.

The world data is growing day by day, and hospitals are slowly adopting big data systems [2]. Applying data analysis in the medical sector is giving excellent benefits. It improves the result and reduces cost. Effective implementation of machine learning improves physicians' work and increases the productivity of the healthcare service. In diagnosing clinical data, significant improvement has been observed using machine learning. Several diseases have been predicted using machine learning like diabetes, heart disease, breast cancer [3, 4]. Machine Learning is closely linked to both statistics and decision-making. They can be used for several purposes, such as forecasting the amount of product sold, forecasting covid cases in the upcoming month, the probability of rainfall occurring in a particular area, selling airline tickets, etc. [5, 6]. As the medical sector has a large dataset, these existing data will help the researchers diagnose the disease early by systematically analyzing data [7]. We have analyzed the binary and multi-classification of heart disease in this research using existing data.

The primary purpose of this study is to classify patients with heart disease using medical records. The five-class classification model can also predict the severity stage of patients with heart disease. We have used six different machine learning algorithms to classify cardiovascular disease in our proposed approach. First, the data was pre-processed. Then the individual algorithms were trained with the same dataset. The weighted sum rule was applied to the decision score provided by the trained algorithms. The fusion models used the new score to classify disease. We have developed one fusion model from two individual algorithms. So in this work, three weighted score fusion models were generated, which provided an improving performance compared to the previous separate machine learning algorithms. A machine learning-based predictive system can reduce physicians' pressure, and a fusion approach will help diagnose disease more efficiently.

2 Related Work

Nowadays, many researchers around the world are focusing on machine learning algorithms in the health field to forecast different diseases. And the use of machine learning in the medical sector has given a notable change in the performance of treatment. This section discusses the various diseases such as diabetes, cardiovascular, kidney, which have been diagnosed using machine learning algorithms.

Comparative analysis of various diseases has been done using machine learning algorithms [8, 9, 10, 11, 12, 13]. In [8], a comparative analysis was developed using machine learning algorithms. Both cardiovascular and diabetes disease datasets were used for classification. Different machine learning algorithms like XGBoost, random forest, and weighted ensemble models were used to predict disease. The essential features that contribute most to the dataset were identified. At last, the performance parameters had been shown to observe the results. Ensemble models showed slightly better accuracy than other models. The researchers also proposed to apply the model in a real-world scenario to check the risk of the disease occurring. They used ten-fold cross-validation, and its accuracy was slightly low. So to increase the accuracy, we have used a fusion of two algorithms rather than a single algorithm.

Two data mining algorithms have been used [14] for the diagnosis of cardiovascular disease. Among them, the SVM showed a strong efficiency. The specificity of SVM and ANN was 74.42% and 73.64%, respectively. Here performance for both algorithms was poor. Using the same dataset, we have been able to get a higher accuracy for every algorithm. And after fusion, the accuracy further improved in our work. Decision-making architectures was introduced in [15, 16, 17, 18]. In [15] a computer-based decision-making architecture was applied with a genetic algorithm. Performance improved significantly after combining the genetic algorithm with an artificial neural network.

Multiple diseases such as cardiovascular, chronic kidney disease (CKD), and diabetes were identified in [7, 19, 20, 21]. Support vector, decision tree, and random forest algorithms were used for classification with a standardized decision support model in [7]. A Chi-square method was employed to select the best features. They implemented SVM (linear, polynomial, and radial) procedures using extracted features. The performance was evaluated with the help of accuracy, specificity, miss classification rate classification parameters, and other parameters. Improved SVM-Radial gave the best accuracy of all the algorithms. We got a better accuracy using weighted score fusion.

A modern gender-based approach was built in [22]. Various classifiers had been applied, and multi-layer perceptron achieved better results than any other data mining techniques. In another study, [23], four data mining algorithms were applied to the diabetes dataset using the WEKA tool. The aim was to evaluate the performance of all algorithms and find out the best one for prediction. The four classifiers were compared, and among them, SVM gave the highest f1-score of 78.2%. The overall performance of SVM was found better than Random Forest, Naive Bayes, and Simple Cart. Survival of heart patients was predicted in [24].

In [25], the data mining technique has been used to diagnose heart disease. KNN and ID3 algorithms were applied. Different accuracy had been observed with the change in the number of attributes. A comparative analysis has been shown in [26] with different machine learning algorithms. Different validation techniques were applied to evaluate the performance. Support vector machine, random forest, decision tree, linear and logistic regression were used for prediction. The highest accuracy achieved by the proposed model was 85.81% using ten-fold cross-validation by random forest.

In [27], KNN imputation was used to handle missing values, and gradient boosting, logistic regression, and naive Bayes had been used to predict diabetes disease. Accuracy, specificity, and sensitivity were measured for performance measurement and validation. Gradient boosting gave the best result for prediction. A computer-based support tool was introduced for diabetics prediction in this paper [28]. Here two data mining algorithms, SVM and Naive Bayes were applied in Matlab. The individual decision was taken from both algorithms, and then the joint outcomes from both algorithms were taken for a new hybrid model. The goal of the hybrid model was to make the system more reliable. It minimized the false-negative response of the system. In a recent study [29],[30] decision level fusion model has been introduced for the classification of heart disease. They have merged two algorithms for classification and obtained a better result than a single algorithm. We used weighted score fusion instead of decision level fusion to get a more improved accuracy.

These studies show that the researchers have used various machine learning algorithms to classify the disease. A limitation of these works is that only binary classification has been done for heart disease classification. It is also essential to determine the result of multi-classification to know what stage a heart disease patient is currently present. So we have classified both two and five-class classifications. And, there is a scope for further improvement in accuracy in the discussed methods. That is why scientists are developing fusion models to classify the disease more accurately. In our work, three weighted fusion models have been created to observe if they can give better results than the separate models that have been used in the fusion. The weighted sum rule was implemented to combine the decision of two individual algorithms [31]. And in our output, we have observed a good improvement in almost every parameter of the fusion models. For two and five-class classifications, we built weighted fusion models and got an improving performance.

3 Materials and Methodology

3.1 Proposed Approach

We conducted quantitative research in this case. Since our goal is to predict disease, we used a quantitative method. We used secondary data in our research and solved a real-world problem. Other researchers have also used this data. In our work, the focus is on determining the seriousness of patients' heart disease along with the presence and absence of it. Two classification models (two and five classes) were developed. First, a multi-classification (five class) system was built to measure the disease's severity. And the binary classification only classified the patients with or without heart disease.

Six machine learning algorithms were used, and three fused models were developed from the six ML models. The algorithms were selected based on their performance. It was noticed that algorithms with accuracy near each other tend to give a better result after fusion. In most cases, this strategy was followed. We did not follow the above-mentioned strategy for only two cases as it did not give a satisfactory outcome, so we randomly checked which combination gave better results and went for it. The two cases mentioned above were for fusion model-1 in multi-classification (80:20 split) & fusion model-1 for binary classification (70:30 split).

First, the data have been pre-processed, which means it was cleaned, missing values were replaced by the most frequent strategy, and feature scaling was implemented. After cleaning, the data were divided into test and train sets. Then, we trained the dataset with six individual algorithms. As the dataset was imbalanced for multi-class classification, oversampling technique was used to balance the dataset. Now for fusion, the decision from two algorithms was added using the weighted sum rule. Before merging, standardization was unnecessary as every individual model's output was in the same score range.

The complete proposed model has been shown in Figure 1. The arrangement of fusion models for binary and multi-classification was slightly different.

Figure 2 shows the overall flow chart for heart disease classification. In the first step, data were pre-processed, and the next step was for training and fusion. This is the generalized structure for weighted score fusion, and this approach was used for all the fusion models using different algorithms. The steps mentioned in figure 2 are listed below in detail.

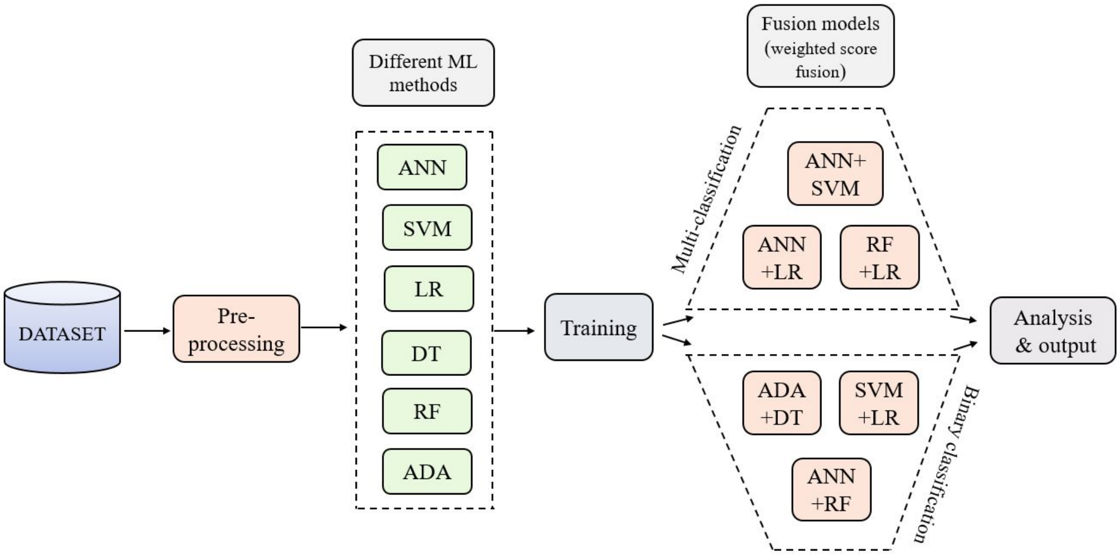


Figure 1: Proposed architecture

3.1.1 Data description

Dataset of the heart disease was taken from the archive of Cleveland UCI [32]. It has 303 cases, 76 attributes, and other missing properties. The most frequent values have been replaced with the missing values in the dataset. Researchers typically use 13 of the 76 attributes. The output contains five labels with no cardiac disease(0) and other values(1-4) reflecting cardiac severity. Categorical values in the dataset were converted into numeric using label encoder. Table 1 below provides a detailed description of each attribute's mean value and standard deviation. It also shows the data distribution of the categorical value in the dataset. The data for training and testing was split into 70:30 and 80:20; for both ratios, output parameters were calculated. Figure 4 shows the data distribution of the continuous value from the dataset.

The output data was balanced for binary classification but unbalanced for multi-class classification. There are some techniques for the imbalanced dataset like oversampling, undersampling, etc. The random oversampling technique has been used for our dataset in multi-class classification. In undersampling, it removes potential valuable data. That is why we have decided to do oversampling. A random set of copies of minority class examples is added to the data in oversampling. That is how it makes the balance between the minority and majority classes. Figure 3a represents the distribution of our training set before making the dataset balanced, and figure 3b is the representation after making them balanced. We first resampled the training dataset using oversampling, and then our model was trained on that dataset. After training, it was used for the prediction of the test dataset. The RandomOverSampler algorithm was applied only on the training dataset. Otherwise, it will create new data in the test set.

3.1.2 Data Transformation

Data needs to be in an appropriate format for the data mining process. That is why data transformation or standardization is necessary. It reduces the redundancy of data. It is the process of changing data structure, configuration, and values of data. Data standardization is a method by which one or more attributes are rescaled to have a mean value of 0 and a standard deviation of 1. In SVM or logistic regression, if one function has a higher value than others, it will dominate other features. So to have the same influence on every attribute, data transformation is essential. It makes data better organized and also improves data quality. Both standardization (z-score) and min-max normalization have been used in this work to transform data for different algorithms. The standard score for a sample x is determined as:

$$q = (x - r)/d \tag{1}$$

where r is the mean of training samples, d is the standard deviation. And the equation for min-max scaler is:

$$h' = \frac{h - min(h)}{max(h) - min(h)} \tag{2}$$

Where h is an original value, and h' is the normalized value.

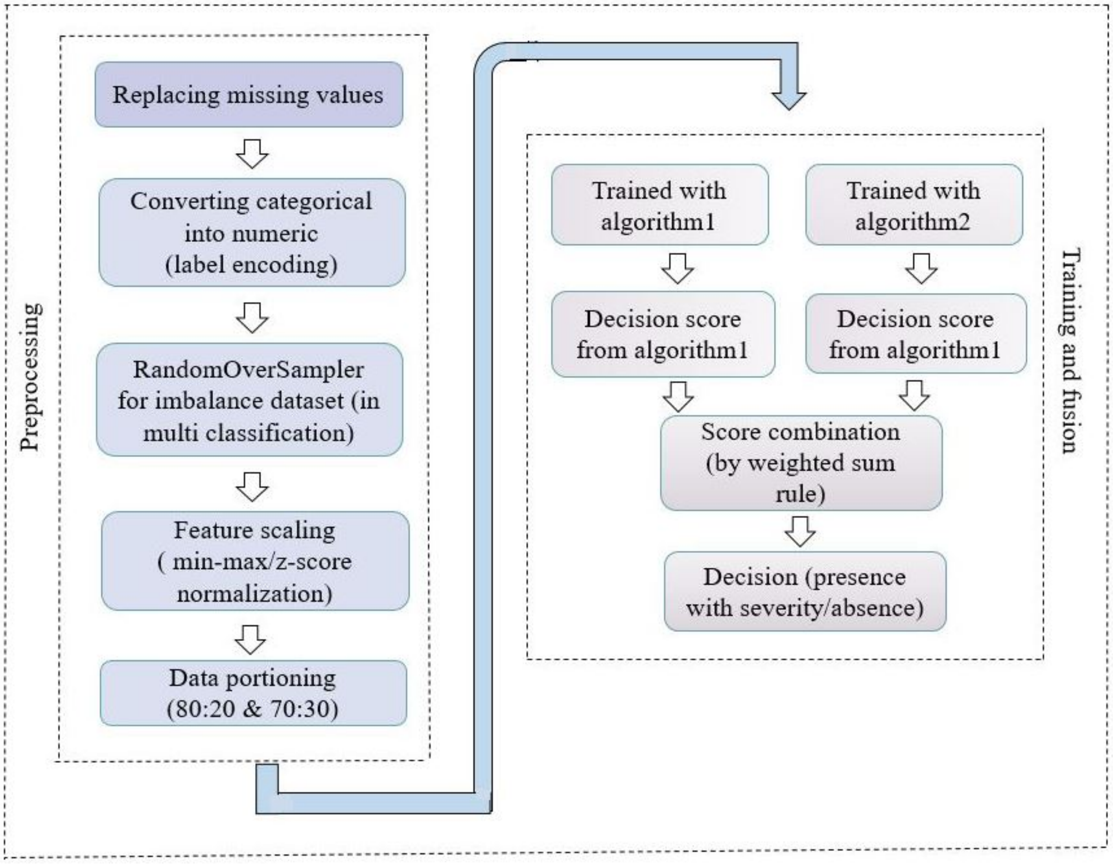
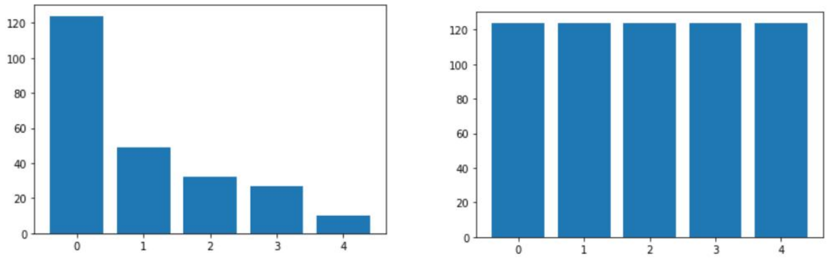

Preprocessing
Replacing missing values
Converting categorical into numeric (label encoding)
RandomOverSampler for imbalance dataset (in multi classification)
Feature scaling (min-max/z-score normalization)
Data portioning (80:20 & 70:30)
Training and fusion
Trained with algorithm1
Trained with algorithm2
Decision score from algorithm1
Decision score from algorithm1
Score combination (by weighted sum rule)
Decision (presence with severity/absence)

Figure 2: Overall flow chart of weighted score level fusion model



(a) Before oversampling

(b) After oversampling

Figure 3: Distribution of target class of training set before and after oversampling

3.1.3 Weighted score fusion

After training all six algorithms, this step was implemented at the decision level. We got a decision probability (Di) for each algorithm to predict the test data. From this decision score, the prediction was made for the test data. Different weights have been assigned to each algorithm's decision probability so that each algorithms' effect varies in the fusion model. If one of the algorithms gave a higher rate for the right decision, we assigned a bigger weight to that and a comparatively smaller weight to the other algorithm. So if one of the algorithms has a weight of .75, then the other will have (1-.75)=.25. We used a loop to check which weights provided the best accuracy for the fusion model and selected them. The sum of the weights used in the fusion model should be 1 for scaling. We selected the weights that gave the best result for the fusion model. The equation for the weighted sum is as follows:

Table 1: Description of Features

S. No.	Attribute	Description	Data distribution	Mean	Std
1	Age	Age in years(29-77)	continuous	54.4	9.07
2	Sex	Gender of patient(1 is represented as male and 0 as female)	0=32.01% 1=67.98%	0.68	0.47
3	Cpt	Chest pain type is classified into 4 values. 1.typical angina 2.atypical angina 3.non-angina pain 4. asymptomatic	1=47.52% 2=16.50% 3=28.38% 4=7.59%	0.97	1.03
4	Thstbps	Resting blood pressure (in mm Hg following hospital admission)with range of (94-200)	continuous	132	17.5
5	S_chol	Serum cholesterol in mm/dl(126-564)	continuous	246	51.7
6	Restelect	Results of resting electrocardiography or ECG is divided into 3 values.0 is represented as normal, 1 as abnormal ST T wave and 2 as the response of left ventricular hypertrophy	0=1.32% 2=48.84% 3=49.83%	0.53	0.52
7	FBS	if level of fasting blood sugar is greater equal 120, represented as 1 for true, 0 for false	0=85.47% 1=15.18%	0.15	0.36
8	thlach	the maximum heart rate that is achieved, in the range between 71 to 202.	continuous	150	22.9
9	Exng	Angina caused by exercise(0 depicted as no and 1 as yes)	0=67.32% 1=32.67%	0.33	0.47
10	Oldpeak	ST depression caused by exercise relative to rest, continuous value in the range between 0 to 6.2	continuous	1.04	1.16
11	Slp	Slope of the peak exercise ST segment	0=6.93% 1=46.20% 2=46.86%	1.4	0.62
12	Ca	Number of major vessels that are colored by floursopy from 0 to 3	0=59.40% 1=21.45% 2=12.54% 3=6.60%	0.73	1.02
13	Thal	Blood disorder represented with three distinct numbers.3 as normal,6 as fixed and 7 as reversible defect	3=5.94% 6=55.44% 7=38.61%	2.31	0.61
14	Output	Classification of heart disease is shown into five values,0 as absence and 1-4 represents the degree of severity	0=54.12% 1-4=45.87%	0.54	0.5

$$D_f = \sum_{i=1}^{n} W_i * D_i \tag{3}$$

Where n is the number of the algorithm used for fusion, we have used two algorithms for every fusion model, so n=2. D_f is the weighted sum, which is the new decision score of the weighted score fusion model. Based on this score, the final decision was given. W_i represents the weight that has been assigned to an algorithm with the decision score, and D_i is the decision probability score for any individual algorithm.

The process of building a fusion model is illustrated below. Two models were used to create a fused model. A weighted score level fusion model was developed by merging those single algorithms, and it worked better than the individual algorithm. The algorithm for weighted level fusion is given below:

After selecting weights, the weighted sum rule was applied at the last step in the algorithm. A weight was assigned to each individual model's decision score in a weighted sum. To select the weights, we used a loop for using various values of weights shown in table 2, and the weights that gave the highest decision score were selected. For example, there were two models in our approach. The individual result was fused, but the outcome of the two algorithms was not taken equally. Rather than weights were assigned that decided the effect of any algorithm in the fusion model.

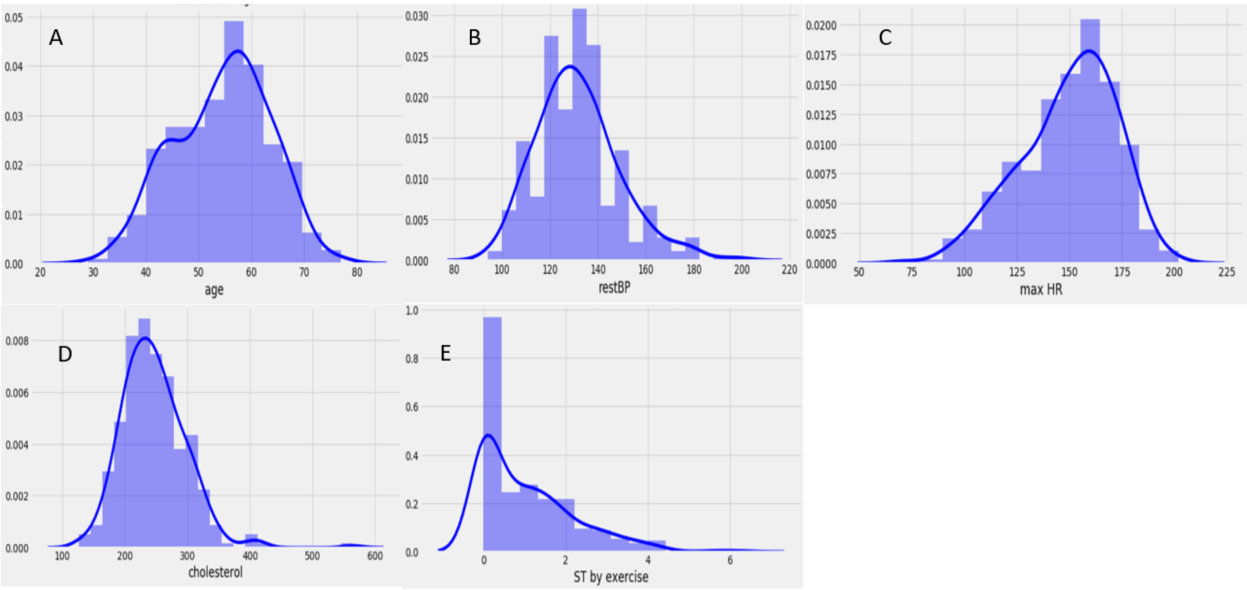


Figure 4: Data distribution for continuous attributes A) Distribution of age B) Resting blood pressure C) Maximum heart rate D) Cholesterol E) Oldpeak

Algorithm 1: Algorithm for weighted score fusion

Input : Two float value D_i,W_i, one int n. D_i is the individual Algorithm's decision score after training,W_i is the corresponding weights of the decision scores, n is the number of separate algorithms in fusion

Output : D_f,new decision score of weighted score fusion model

```
$W_1 = 1$
for $i \leftarrow 0$ to $19$ do
    $D_f = 0$
    $W_1 = W_1 - 0.05$, $W_2 = 1 - W_1$ ;          // $\sum_{i=1}^{n} W_i = 1$
    for $i \leftarrow 0$ to $n$ do
        $D_f = \sum W_i * D_i$
    end
end
Select the weights $(W_1, W_2)$ that gave the highest decision score.
$D_f = \sum W_i * D_i$ ;          // Final fused decision score using selected weights
```

Then from the new decision score, the result of our fusion model was predicted. The code is given here. The steps for the fusion is given below in detail:

1. The output of any algorithm was determined from the prediction scores. It ranges between 0 to 1. At first, the prediction scores ($D1$ & $D2$) of the two algorithms were taken.
2. Then using the selected weights($W1$ & $W2$), the final decision score (Df) was obtained using equation 3. From this score, we got our output value. If the final decision score was greater than .5, the output was zero else one for binary classification.
3. A for loop was used to select the weights from the different combinations shown in the table 2. The final decision score (Df) of the test set was calculated using the different combinations of weights ($W1$ & $W2$) and using the two algorithms' given decision probability scores ($D1$ & $D2$). The weights that showed the highest decision score (Df) were selected for fusion among the combinations.
4. After selecting the weights, they were assigned to the decision scores of the individual models like in the equation 3, and the fusion decision score was calculated for both binary and multi-class classification.

5. The output was calculated from this final decision score (Df). There was a decision score for every test data by which the prediction was made.
    (a) The new decision score (Df) of the fusion model contained two different float numbers for every test data in binary class, and the sum of the values was one. The index of the higher value indicates the target class of that test data. The prediction was made by extracting the index number of the greater value.
    (b) Each algorithm's prediction score contained five float numbers representing the five different classes for multi-class classification. The final decision score had five different values in an array for every test data. The index containing the maximum value was the target class. We extracted the index number from the array to know the predicted output.
6. At last, the prediction was made for the test data from the final decision score (Df).

We used the heart disease dataset in our proposed framework for both binary and multi-classification problems. In multi-classification, data were split into multiple groups where there were two classes in binary classification. Performance parameters were measured to see how well the models performed.

Table 2: Combinations of weights used in fused algorithms

Number of loop	W1	W2	Sum of weights
1	0.95	0.05	1
2	0.9	0.1	1
3	0.85	0.15	1
4	0.8	0.2	1
5	0.75	0.25	1
6	0.7	0.3	1
7	0.65	0.35	1
8	0.6	0.4	1
9	0.55	0.45	1
10	0.5	0.5	1
11	0.45	0.55	1
12	0.4	0.6	1
13	0.35	0.65	1
14	0.3	0.7	1
15	0.25	0.75	1
16	0.2	0.8	1
17	0.15	0.85	1
18	0.1	0.9	1
19	0.05	0.95	1

3.2 Modeling

3.2.1 Logistic regression(LR)

Logistic regression is a specific form of GLM, which is frequently abbreviated as a Generalized Linear Model. It is like linear regression, but it predicts true or false. Instead of adjusting a line to the data, LR provides a s-shaped logistic function. The output probability for any given problem is found from the curve, and it is generally used for binary classification. Logistic regression can work with both continuous and discrete data. In linear regression, the line is fit by least squares, whereas logistic regression uses maximum likelihood. The likelihood is calculated for different curves, and the curve with maximum likelihood is selected for classification. Logistic regression determines the probability of the binary response.

The line for linear regression is selected in such a way that the distance between the summation of all points and lines should be minimum. The equation for the plane is written as:

$$y = \omega^T x + b \tag{4}$$

The trouble with linear regression is that we need to adjust the best fit line whenever new data comes. The exact coefficient is needed to find that will understand which is the best match line for that model, and then those particular coefficients or slopes will be adjusted to achieve the optimal plane. If x is the data point, then $\omega^T x$ is the distance between a particular data and plane, considering b=0.

Distance between data point and the particular plane is : $\sum_{i=1}^{n} \omega^T x_i$

Classes are classified according to the equation:

$$y_i \omega^T x_i \geq 0 \quad \text{correctly classified} \tag{5}$$

$$y_i \omega^T x_i \leq 0 \quad \text{incorrectly classified} \tag{6}$$

To get the best fit line, cost function($\sum_{i=1}^{n} y_i \omega^T x_i$) should be maximum. Here, ω^T is the coefficient which needs to update until maximum summation is found. The impact of an outlier has been avoided by adding the sigmoid function in the equation.Then the equation becomes:

$$max \sum_{i=1}^{n} f(y_i \omega^T x_i) \tag{7}$$

The sigmoid function is denoted as f in this equation.

In logistic regression, we tuned the parameter C, the regularization strength. For our proposed model, we used C = 1.

3.2.2 Support vector machine(SVM)

SVM or support vector machine is a supervised machine learning algorithm like linear regression, but here hyperplane with maximum marginal distance is selected to separate data into their categories.[13] SVM classifies positive and negative classes, and it has been shown in 8 and 9:

$$\mathbf{x}_i \mathbf{w} + b \geq +1, \quad y_i = +1 \tag{8}$$

$$\mathbf{x}_i \mathbf{w} + b \leq -1, \quad y_i = -1 \tag{9}$$

$$\equiv \tag{10}$$

$$y_i(\mathbf{x}_i \mathbf{w} + b) - 1 \geq 0, \qquad \forall i \tag{11}$$

Here x and w are vectors, and bias is denoted by b. In SVM, it is important to optimize the marginal distance, which means the value of w and b must be minimized to get the largest marginal distance $||w||$. By optimizing the value of w and b, we can get our required distance to select the most fitted hyperplane.

$$min_{w}{,_b}\,{,_\xi} = \frac{1}{2}\|w\|^2 \tag{12}$$

However, real data will have many overlaps, and if we follow the equation 13, then there will be over-fitting. Therefore a small amount of error is added to avoid over-fitting.

$$min_{w}{,_b}\,{,_\xi} = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} \xi_i \tag{13}$$

Here, C is the number of errors that the model will consider, and ξ is the value of error. A larger value of C indicates more error rate; if C is lower, there could be fewer errors [33]. In our model, C was 1 and 10 respectively for binary and multi-class.

3.2.3 Decision tree(DT)

A decision tree is a tree-like arrangement that is built based on attributes. The very top of the tree is called the root node or the root. Internal nodes are the nodes between the root and leaf node. Leaf nodes are the final class labels that represent the outcome. While constructing a decision tree, there is a concept or algorithm, which is ID3 algorithm. This algorithm tells that the first step for creating a decision tree is to select the right attributes for splitting the tree. To determine the best separation, we need to measure the impurity. A very popular model to calculate impurity is gini.

Entropy is another model to estimate impurity by selecting which feature should be the root node. Choosing the right attribute for the root node will help to find the leaf node quickly and reduce time. A decision tree aims to get the leaf node quickly by selecting the right parameters. The equation for both gini and entropy is:

$$Gini : Gini(E) = 1 - f(x) = 1 - \int_{j=1}^{n} Pj^2 \tag{14}$$

$$Entropy : H(E) = 1 - f(x) = - \int_{j=1}^{n} Pj \log_2 Pj \tag{15}$$

Information gain calculates the total entropy value from the top to the bottom node of the decision tree. The highest value of information gain is selected to construct the decision tree. Entropy is the calculation for one node, and information gain combines all the entropy. Entropy ranges between 0 to 1. 1 means the subset is completely impure where 0 is a pure subset. The construction of a DT is simple, and the implementation process is easy. The creation of a new node continues until a ground condition has not been formed.[34].

Hyperparameter tuning is necessary for the decision tree. It is a value search method that will optimize the model. Our proposed model's tuned parameters were maximum depth, the minimum number of sample leaves, and maximum features. This stopping criterion of these parameters helps to reduce over-fitting. In our decision tree model, maximum depth and maximum features were 8, the minimum sample leaf number was seven, and gini was used as the criterion.

3.2.4 Random forest(RF)

The ensemble method combines multiple models to create an optimal model, and it is a machine learning technique. There are two forms of the ensemble model. Bagging and boosting. Random forest is a bagging technique. Bagging is also known as the bootstrap aggregating method. It has two main principles: row sampling and voting classifier. The given records are resampled and passed to the next base learner models to train. Aggregating is the voting classifier concept where, after training, output for test data will be selected for that class, which has the maximum vote from base learner models [35]. Figure 5 shows a generalized model for the random forest.

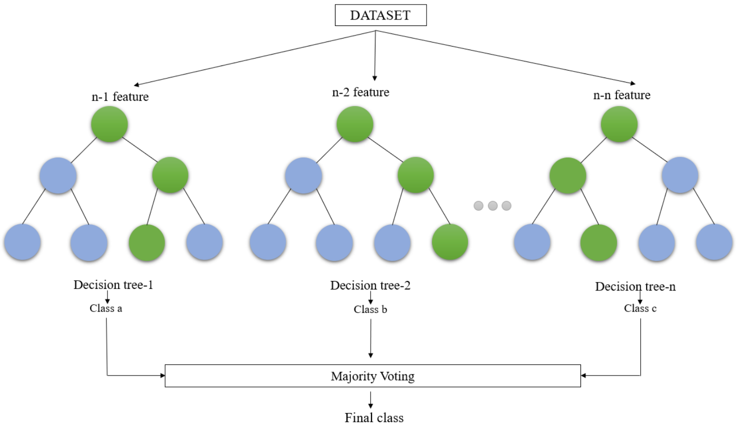


Figure 5: Generalized structure for random forest

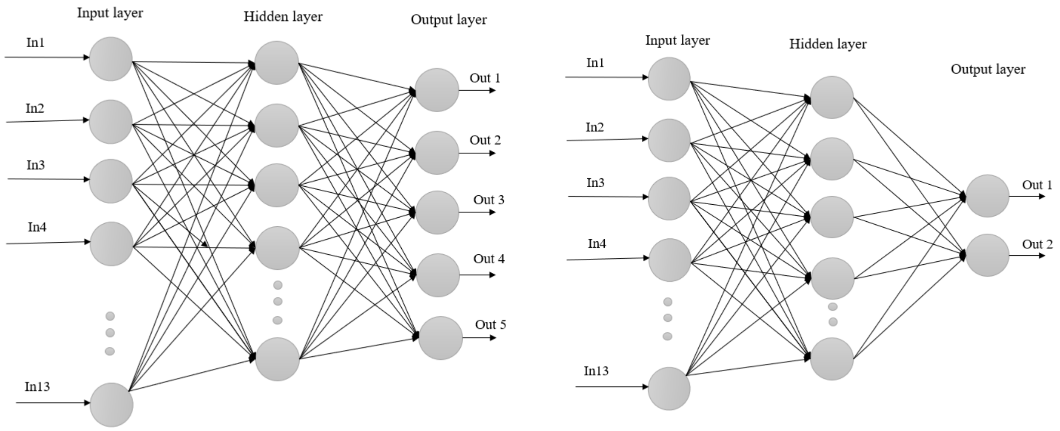
A decision tree is based on a complete dataset, including all features and variables, where the random forest randomly chooses some row and unique features from the dataset to create several trees and then combines the result; the decision tree has two properties.

- Low bias: Since DT is built in its complete depth, training error is significantly less.
- High variance: The inability to classify test data leads to a larger amount of error.

In a random forest, the high variance is converted into low variance due to the use of multiple decision trees. Adding new records in the dataset does not have a high impact on performance, so the accuracy does not decrease easily. In the regression problem, the random forest takes the mean or median of all decision trees, and in classifiers, it takes the majority vote for prediction. The number of estimators in our work was 100.

3.2.5 Artificial neural network(ANN)

An artificial neural network is built on brain structure. Deep learning is a subfield of machine learning that uses algorithms inspired by the structure and function of brains. Such models used in deep learning are named artificial neural networks. So, artificial neural networks are computational structures that are inspired by the human brain. These neural networks are a collection of connected units called artificial neurons or only neurons. Each neuronal connection can transmit signals from one neuron to the next [29]. Typically neurons are organized in layers, and different layers perform different kinds of transformations depending on their inputs. Signals basically move from the first layer, called the input layer, to the last layer, called the output layer. Each layer between the input and the output layer is considered as a hidden layer. Figure 6 shows the structure for both binary and multi-classification of our model. The difference between them is that in multi-classification, the output layer contains five nodes, one for the absence of heart disease and the other four for reflecting the severity of the heart disease. On the other hand, the binary classification structure has two nodes for output.



(b) Binary classification

(a) Multi-classification(five class)

Figure 6: Architecture of artificial neural network

Here, the dense layer was used to build the model. The first layer consisted of 13 neurons in our proposed model. Each of the nodes in the input layer reflects the individual features of each sample from the dataset. A weight has been allocated to each connection between the nodes. In training, the target tries to optimize the model's weights. The weights were continually updated to reach their optimal values. This optimization depends on the optimizer. We used stochastic gradient descent (SGD) in our model as the optimizer. SGD minimizes a given loss function. It assigns weights in such a way to make the loss function as close to zero as possible. The other hyper-parameters were batch size, learning rate, regularization. Batch size represents how many pieces of data we want to send to the model at once. The learning rate controls the amount of change in the model to calculate the error during the model weight update. Over-fitting occurs when the model is good at training samples but gives poor performance for testing. To prevent it, l2 regularization was used. Our artificial neural network contains one hidden layer for both binary and multi-classification. The number of neurons in the hidden layer was 8, and SGD was used as the optimizer.

3.2.6 AdaBoost(ADA)

In an adaptive boosting approach [36], decision stump [37] is used as a base learner. A stump is a tree with one node and two leaves, which is selected based on the entropy of the data given. The stump with the lowest entropy is chosen for this method. AdaBoost has three main characteristics.

- To make a classification, AdaBoost combines a lot of weak learners (always stumps) to produce a strong algorithm.
- Some stumps get more say in classification than others.
- Each stump is made by taking the previous stump's mistakes into account.

The error made by the first base learner will be passed to the next base learner, and it will continue until the number of base learners that need to be created is declared. Boosting iterations are aimed at reducing the error of classification of the combined classification model over the training set. For classification, the maximum number of estimators was 200, and a .01 learning rate was used.

3.2.7 Hyperparameter optimization

We have optimized the six algorithms' different parameters. The tuning of the parameters was different for both binary and multi-class classification. All optimal parameters have been listed in table 3.

Table 3: Optimal hyperparameters used in algorithms

Algorithms	**Optimal Parameters**			
	Binary class		Multi class	
	train-test ratio 70:30	train-test ratio 80:20	train-test ratio 70:30	train-test ratio 80:20
Logistic regression	C=1	C=1	C=0.1	C=0.001

Support vector machine	C=1
	gamma=.1
		C=1
		gamma=.01
			C=0.01	C=100

Decision tree	criterion=gini maximum depth= 8 maximum features= 8 minimum sample leaf=7 splitter= random	criterion=gini maximum depth= 8 maximum features= 8 minimum sample leaf=7 splitter= random	N/A	N/A
Random forest	number of estimators=100	number of estimators=100	number of estimators=100	number of estimators=200

Artificial neural network	learning rate=0.01
	optimizer=SGD
	batch size=10
	epoch=15
		learning rate=0.01
		optimizer=SGD
		batch size=10
		epoch=15
			learning rate=0.01
			optimizer=SGD
			batch size=5
			epoch=20
				learning rate=0.01
				optimizer=SGD
				batch size=5
				epoch=20

AdaBoost	number of estimators=250 learning rate =.01	number of estimators=200 learning rate =.01	N/A	N/A

3.3 Performance Parameter

Classification model performance is measured with the term of accuracy, precision, recall, and F1-score. The model's performance can be interpreted from the value of these parameters. Here,

- True positive (T_P): Outcome is positive, and prediction is positive.
- True negative (T_N): Outcome is negative, and prediction is negative.
- False positive (F_P): Outcome is negative, but prediction is positive.
- False negative (F_N): Outcome is positive, but prediction is negative.

Accuracy is the measure of the percentage of correctly classified objects:

$$Accuracy = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} * 100 \tag{16}$$

Precision is also referred to as the false-positive rate. From precision, we get the number of correctly classified observations as positive to the total classified positive observations.

$$Precision = \frac{T_P}{T_P + F_P} \tag{17}$$

Recall is often referred to as a truly positive rate. It is the ratio of total positive assumptions and the total amount of positive class attributes.

$$Recall = \frac{T_P}{T_P + F_N} \tag{18}$$

F1-score is the weighted average of precision and recall. When there is a need for the balance of precision and recall, it is required to take both the false positives and the false negatives into account. For an uneven class distribution, F1-score is generally more useful than accuracy.

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{19}$$

ROC AUC score is also calculated as a classification parameter. It computes the area under the receiver operating characteristics curve using prediction scores.

To illustrate the classifiers' performance, a confusion matrix has been used for the binary classification model. It summarizes the results of the predictions of a model. The matrix is given in table 4.

Table 4: Confusion matrix

	Actually Positive	Actually Negative
Predicted Positive	True Positive T_P	False Positive F_P
Predicted Negative	False Negative F_N	True Negative T_N

4 Result and Discussion

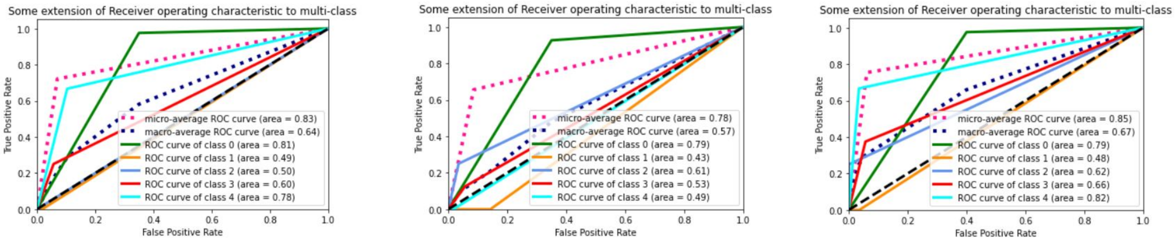
Six algorithms were applied to the dataset. A grid search method was carried out to find the best-fit parameters for the algorithms. After training with those algorithms, three new fused models were developed using those models' decision scores. A notable improvement was found in the result of all fused models. The key reason behind the fused model's improvement is that if there is a miss classification in one algorithm, there is a probability that the other algorithm may classify that particular data correctly. So after merging, there is a chance that we get the correct result for that specific case. This concept assists in giving the fused model a better efficiency.

The macro average was taken for the value of precision, recall, and F1-score in binary classification, and for multi-classification, the value of the weighted average was taken.

Table 5: Fusion Model-1 with logistic regression and random forest for multi classification

Ratio	**Algorithm**	**Acc**	**Precision**	**Recall**	**F1-score**
70:30	LR	59.34	63	59	61
	RF	59.34	54	59	56
	LR+RF	**65.93**	**66**	**66**	**66**
80:20	LR	72.13	64	72	67
	RF	65.57	62	66	63
	LR+RF	**75.41**	**72**	**75**	**71**

The result of multi-classification has been displayed in table 5,6 and 7. Table 5 shows the performance of LR, RF, and the combined model of them. After merging the two models' decisions, the fused models' performance parameters increased (LR+RF). It gave the highest accuracy of 75.41% among the three fusion models.



(a) LR

(b) RF

(c) LR+RF

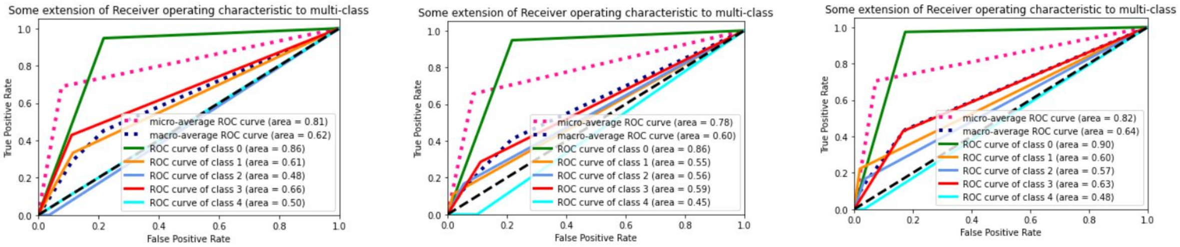
Figure 7: ROC curve of Model-1 for multi-classification(ratio 80:20)

Figure 7 displays the Roc curve of logistic regression, random forest, and the fusion model(LR+RF) for five class classifications.

Table 6: Fusion Model-2 with support vector machine and artificial neural network for multi classification

Ratio	Algorithm	Acc	Precision	Recall	F1-score
70:30	SVM	64.84	65	65	63
	ANN	65.93	70	66	62
	SVM+ANN	**67.03**	**65**	**67**	**63**
80:20	SVM	65.57	65	66	64
	ANN	68.85	72	69	70
	SVM+ANN	**72.13**	**75**	**72**	**71**

Table 6 has represented the outcome of algorithm SVM and ANN. This fusion model provided 72.13% accuracy, and the increment in the accuracy was 4% and 7% from ANN and SVM, respectively, at 80:20 split.



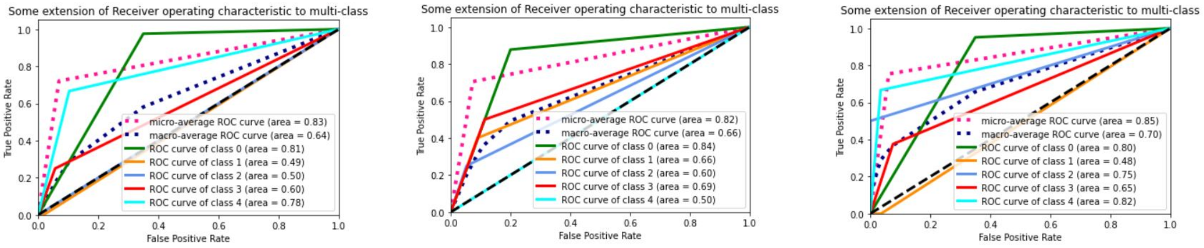
(a) ANN (b) SVM (c) SVM+ANN

Figure 8: ROC curve of Model-2 for multi-classification(ratio 80:20)

The Roc curve of fusion model-2 and SVM and ANN curves have been displayed in figure 8.

Table 7: Fusion Model-3 with artificial neural network and logistic regression for multi classification

Ratio	Algorithm	Acc	Precision	Recall	F1-score
70:30	ANN	65.93	60	66	62
	LR	62.64	62	63	62
	ANN+LR	**67.03**	**62**	**67**	**64**
80:20	ANN	70.49	70	70	70
	LR	72.13	64	72	67
	ANN+LR	**75.41**	**72**	**75**	**73**



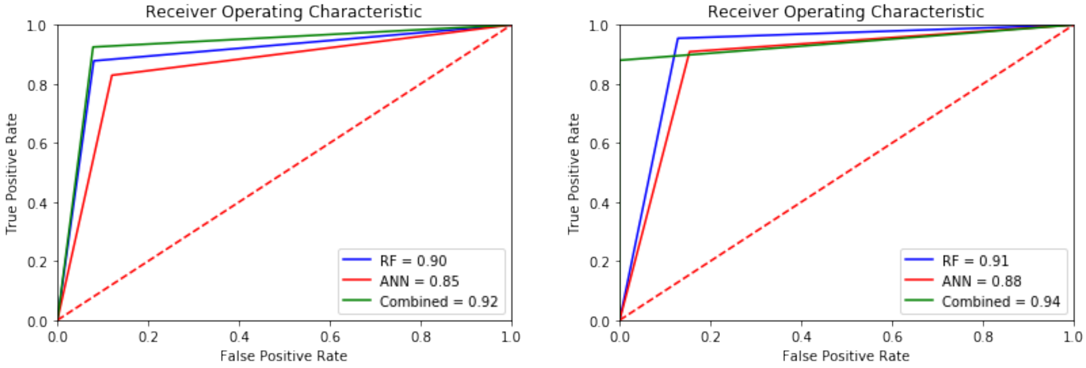
(a) LR (b) ANN (c) ANN+LR

Figure 9: ROC curve of Model-3 for multi-classification(ratio 70:30)

In table 7, an accuracy of 75.41% was obtained by fusion model-3, which was also the highest. But the precision did not improve for the 80:20 ratio of the hybrid model-3 (ANN+LR). As regards medical diagnosis, a false-negative is considered a criminal offense. This is why we put more focus on recall than on precision. The accuracy is 75.41%. After merging, recall improved along with accuracy and f1-score. In figure 9, the roc curve of this model has been displayed. So these fusion systems provided a good performance overall. This comparison has been displayed in figure 13. Four performance parameters were compared for the three multi-classification models in the figure.

Table 8: Fusion Model-1 with artificial neural network and random forest for binary classification

Ratio		**Tp**	**Fp**	**Acc**	**Prc**	**Recall**	**F1-score**	**Roc-Auc score**
		Fn	**Tn**					
70:30	ANN	48	2	85.71	87	86	85	84.59
		11	30					
	RF	47	3	91.21	91	91	91	90.90
		5	36					
	ANN +RF	**48**	**2**	**93.41**	**93**	**93**	**93**	**93.59**
		4	**37**					
80:20	ANN	35	2	91.80	92	91	91	91.10
		3	21					
	RF	34	3	93.44	93	94	93	93.86
		1	23					
	ANN +RF	**35**	**2**	**95.08**	**95**	**95**	**95**	**94.61**
		1	**23**					



(a) 70:30 ratio

(b) 80:20 ratio

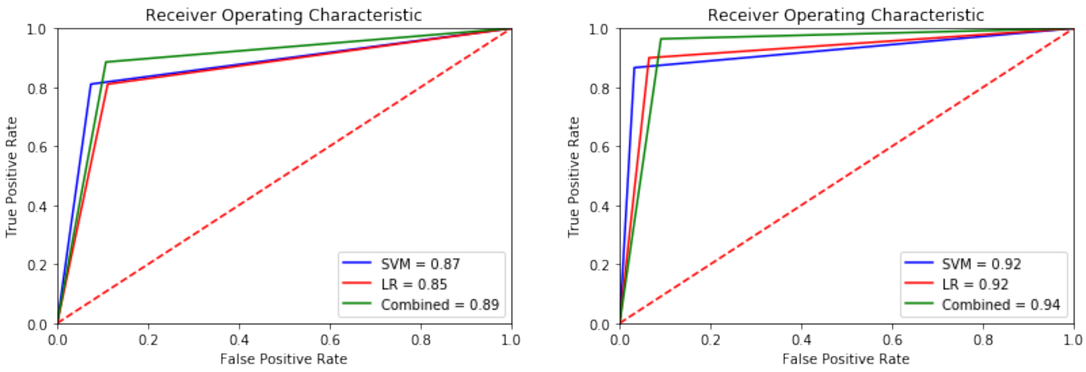
Figure 10: ROC curve of Model-1 for binary classification

Table 8,9, and 10 has the output of the binary classification. The performance parameters and the confusion matrix have been shown in these tables. Here we have gained the highest accuracy of 95.08% from both fusion model-1(ANN+RF) and model-3(ADA+DT). Fusion model-1 has been displayed in table 8.

Table 9: Fusion Model-2 with support vector machine and logistic regression for binary Classification

Ratio		**Tp**	**Fp**	**Acc**	**Prc**	**Recall**	**F1-score**	**Roc-Auc score**
		Fn	**Tn**					
70:30	SVM	50	4	87.91	88	87	87	86.84
		7	30					
	LR	48	6	85.71	85	85	85	84.98
		7	30					
	SVM +LR	**50**	**4**	**89.90**	**89**	**88**	**89**	**88.93**
		6	**31**					
80:20	SVM	34	1	91.80	93	91	91	90.88
		4	22					
	LR	32	3	90.16	90	90	90	89.95
		3	23					
	SVM +LR	**33**	**2**	**93.44**	**93**	**93**	**93**	**93.30**
		2	**24**					

Fusion model-2 and 3 also performed well. Table 9 represents model-2(SVM+LR), which gave 93.44% accuracy with an 80:20 split ratio.



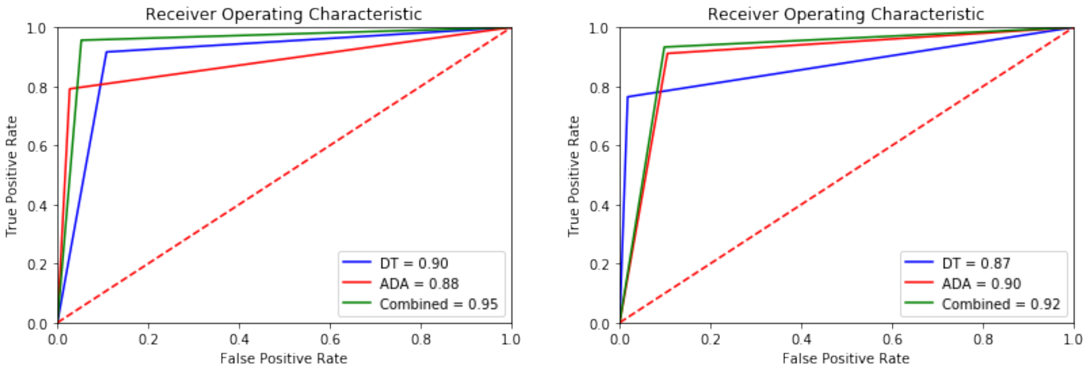
(a) 70:30 ratio

(b) 80:20 ratio

Figure 11: ROC curve of Model-2 for binary classification

Table 10: Fusion Model-3 with adaboost and decision tree for binary classification

Ratio		Tp	Fp	Acc	Prc	Recall	F1-score	Roc-Auc score
		Fn	Tn					
70:30	ADA	51	6	90.11	89	90	90	90.32
		3	31					
	DT	56	1	90.11	92	87	89	87.36
		8	26					
	ADA +DT	**53**	**4**	**92.31**	**92**	**92**	**92**	**91.61**
		3	**31**					
80:20	ADA	36	1	90.16	91	88	89	88.23
		5	19					
	DT	33	4	90.16	89	90	90	90.43
		2	22					
	ADA +DT	**36**	**1**	**95.08**	**95**	**94**	**95**	**95.19**
		2	**22**					



(a) 70:30 ratio

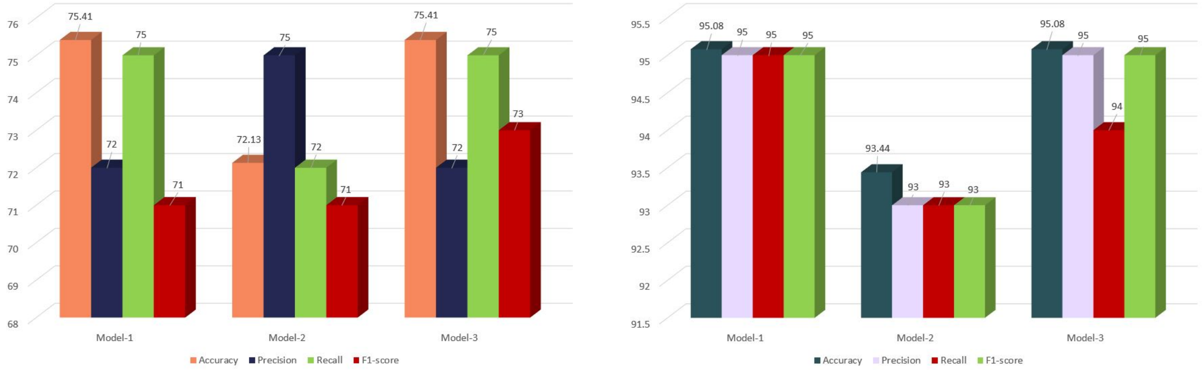
(b) 80:20 ratio

Figure 12: ROC curve of Model-3 for binary classification

The performance parameters of model-3 have been shown in table 10. Here, model-3 provided an accuracy of 92.31% and 95.08% after fusion with the 70:30 and 80:20 split ratio, respectively. After fusion, the accuracy showed improvement compared to ADA and DT.

In figure 10,11, 12, the roc curve for the binary fusion model-1,2 and 3 have been shown, respectively.

Figure 13 shows the comparison chart for the three multi and binary classification models. All the models' efficiency has significantly improved after fusion.



(a) Multi-classification

(b) Binary classification

Figure 13: Comparison of weighted level fusion models (for 80:20 ratio)

In table 11, various approaches have been discussed along with their highest accuracy for diagnosing different diseases. Some of those works used the same dataset we used, and by analyzing them, we can say that our weighted fusion model's performance is quite good comparing those. The research with two-class classification has done more than five classes, so we have discussed the previous work of binary classification of heart disease more than five class classifications.

Table 11: Comparison of various approaches for the diagnosis of diseases

Method	Results	Ref
Random forest, Decision tree, improved SVM kernel	The best result was achieved from improved SVM that was 89.9%; RF gave 82% with an 80:20 split ratio.	[7]
Naive Bayes, multi layer perceptron, bagging multi layer, and other algorithms	The highest accuracy obtained by Naive Bayes was 92% using 55 attributes	[22]
Deep learning, decision tree and others	Deep learning performed best among others with an accuracy of 98.07% for diabetes prediction	[38]
Logistic regression, random forest and others	It gave 83% accuracy using all features	[39]
Support vector machine, logistic regression	SVM & LR gave 85% & 87% accuracy respectively with Cleveland dataset (using 80% data as training)	[40]
Support vector machine, multilayer perceptron neural network	Accuracy for multi classification was 68.86% from SVM using 80% data for training (Cleveland dataset)	[41]
Weighted fuzzy diagnostic system	This method gave 92.31% accuracy for Cleveland's heart disease dataset using 70:30 train test ratio	[42]
Hybrid method (ANN Fuzzy AHP)	Provided 91.10% accuracy	[43]

5 Conclusion

A significant percentage of the world's population is struggling with heart disease. To identify patients with heart disease, machine learning is necessary. This paper has focused on implementing a fused prediction model to identify patients with and without heart disease using machine learning algorithms. Binary classification classifies only the presence and absence of heart disease. In multi-classification, if heart disease is present, the severity can also be predicted. Our proposed models did both binary and multi-class classifications of heart disease. Multiple hybrid classification models have been designed to diagnose the disease, and then the distinction was made among them. The fusion models reduced the risk factors and improved the output in terms of accuracy, efficiency, and other factors. We saw a good improvement in the fusion model in both classifications, which makes the model more reliable. The decision from each algorithm was taken, and using the weighted sum rule, the scores were merged to form a new score. This new score is the decision score of our fusion model. Based on this score, the output will be predicted. The accuracy obtained for multi-classification from the fusion model is between 72-75%, and for binary classification, the result is 93-95%. After fusion, the performance improvement compared to the individual algorithm was about 1-3%. This increase happened to all the parameters of the evaluation matrix.

A future real-world application of such work can be a mobile application, a web-based platform for physicians to update patient information. Patients would know the result with its risk factors easily. Depending on the severity, patients suffering from illness could be treated. This weighted fusion approach could also apply to other diseases as well.

6 Code and Data Availability

The Github link for the code and data are available here.

References

[1] Nathan D Wong. Epidemiological studies of chd and the evolution of preventive cardiology. *Nature Reviews Cardiology*, 11(5):276, 2014.

[2] David Gans, John Kralewski, Terry Hammons, and Bryan Dowd. Medical groups' adoption of electronic health records and information systems. *Health affairs*, 24(5):1323–1333, 2005.

[3] Meriem Amrane, Saliha Oukid, Ikram Gagaoua, and Tolga Ensari. Breast cancer classification using machine learning. In *2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT)*, pages 1–4. IEEE, 2018.

[4] Sellappan Palaniappan and Rafiah Awang. Intelligent heart disease prediction system using data mining techniques. In *2008 IEEE/ACS international conference on computer systems and applications*, pages 108–115. IEEE, 2008.

[5] Hafsa Binte Kibria, Oishi Jyoti, and Abdul Matin. Forecasting the spread of the third wave of covid-19 pandemic using time series analysis in bangladesh. *Informatics in Medicine Unlocked*, page 100815, 2021.

[6] Sam Cramer, Michael Kampouridis, Alex A Freitas, and Antonis K Alexandridis. An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives. *Expert Systems with Applications*, 85:169–181, 2017.

[7] Karthikeyan Harimoorthy and Menakadevi Thangavelu. Multi-disease prediction model using improved svm-radial bias technique in healthcare monitoring system. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–9, 2020.

[8] An Dinh, Stacey Miertschin, Amber Young, and Somya D Mohanty. A data-driven approach to predicting diabetes and cardiovascular disease with machine learning. *BMC medical informatics and decision making*, 19(1):211, 2019.

[9] Archana Singh and Rakesh Kumar. Heart disease prediction using machine learning algorithms. In *2020 international conference on electrical and electronics engineering (ICE3)*, pages 452–457. IEEE, 2020.

[10] Md Mamun Ali, Bikash Kumar Paul, Kawsar Ahmed, Francis M Bui, Julian MW Quinn, and Mohammad Ali Moni. Heart disease prediction using supervised machine learning algorithms: performance analysis and comparison. *Computers in Biology and Medicine*, 136:104672, 2021.

[11] Devansh Shah, Samir Patel, and Santosh Kumar Bharti. Heart disease prediction using machine learning techniques. *SN Computer Science*, 1(6):1–6, 2020.

[12] Manoj Diwakar, Amrendra Tripathi, Kapil Joshi, Minakshi Memoria, Prabhishek Singh, et al. Latest trends on heart disease prediction using machine learning and image fusion. *Materials Today: Proceedings*, 37:3213–3218, 2021.

[13] Hafsa Binte Kibria, Abdul Matin, Nusrat Jahan, and Sanzida Islam. A comparative study with different machine learning algorithms for diabetes disease prediction. In *2021 18th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pages 1–8. IEEE, 2021.

[14] Haleh Ayatollahi, Leila Gholamhosseini, and Masoud Salehi. Predicting coronary artery disease: a comparison between two data mining algorithms. *BMC public health*, 19(1):1–9, 2019.

[15] Zeinab Arabasadi, Roohallah Alizadehsani, Mohamad Roshanzamir, Hossein Moosaei, and Ali Asghar Yarifard. Computer aided decision making for heart disease detection using hybrid neural network-genetic algorithm. *Computer methods and programs in biomedicine*, 141:19–26, 2017.

[16] Aditi Gavhane, Gouthami Kokkula, Isha Pandya, and Kailas Devadkar. Prediction of heart disease using machine learning. In *2018 second international conference on electronics, communication and aerospace technology (ICECA)*, pages 1275–1278. IEEE, 2018.

[17] Monther Tarawneh and Ossama Embarak. Hybrid approach for heart disease prediction using data mining techniques. In *International Conference on Emerging Internetworking, Data & Web Technologies*, pages 447–454. Springer, 2019.

[18] Saman Iftikhar, Kiran Fatima, Amjad Rehman, Abdulaziz S Almazyad, and Tanzila Saba. An evolution based hybrid approach for heart diseases classification and associated risk factors identification. *Biomedical Research*, 28(8):3451–3455, 2017.

[19] Jing Xiao, Ruifeng Ding, Xiulin Xu, Haochen Guan, Xinhui Feng, Tao Sun, Sibo Zhu, and Zhibin Ye. Comparison and development of machine learning tools in the prediction of chronic kidney disease progression. *Journal of translational medicine*, 17(1):1–13, 2019.

[20] Md Kamrul Hasan, Md Ashraful Alam, Dola Das, Eklas Hossain, and Mahmudul Hasan. Diabetes prediction using ensembling of different machine learning classifiers. *IEEE Access*, 8:76516–76531, 2020.

[21] Sibo Qiao, Shanchen Pang, Gang Luo, Silin Pan, Zengchen Yu, Taotao Chen, and Zhihan Lv. Rlds: An explainable residual learning diagnosis system for fetal congenital heart disease. *Future Generation Computer Systems*, 128:205–218, 2022.

[22] Mofreh A Hogo. A proposed gender-based approach for diagnosis of the coronary artery disease. *SN Applied Sciences*, 2:1–17, 2020.

[23] Ayman Mir and Sudhir N Dhage. Diabetes disease prediction using machine learning on big data of healthcare. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6. IEEE, 2018.

[24] SM Mehedi Zaman, Wasay Mahmood Qureshi, Md Mohsin Sarker Raihan, Ocean Monjur, and Abdullah Bin Shams. Survival prediction of heart failure patients using stacked ensemble machine learning algorithm. *arXiv e-prints*, pages arXiv–2108, 2021.

[25] J Thomas and R Theresa Princy. Human heart disease prediction system using data mining techniques. In *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pages 1–5. IEEE, 2016.

[26] Yeshvendra K Singh, Nikhil Sinha, and Sanjay K Singh. Heart disease prediction system using random forest. In *International Conference on Advances in Computing and Data Sciences*, pages 613–623. Springer, 2016.

[27] Roshan Birjais, Ashish Kumar Mourya, Ritu Chauhan, and Harleen Kaur. Prediction and diagnosis of future diabetes risk: a machine learning approach. *SN Applied Sciences*, 1(9):1112, 2019.

[28] Zhilbert Tafa, Nerxhivane Pervetica, and Bertran Karahoda. An intelligent system for diabetes prediction. In *2015 4th Mediterranean Conference on Embedded Computing (MECO)*, pages 378–382. IEEE, 2015.

[29] Hafsa Binte Kibria and Abdul Matin. An efficient machine learning-based decision-level fusion model to predict cardiovascular disease. In *International Conference on Intelligent Computing & Optimization*, pages 1097–1110. Springer, 2020.

[30] Hafsa Binte Kibria, Abdul Matin, and Sanzida Islam. Comparative analysis of two artificial intelligence based decision level fusion models for heart disease prediction. In *International Semantic Intelligence Conference*, volume 2786, pages 314–322. CEUR Workshop Proceedings, 2020.

[31] Abdul Matin, Firoz Mahmud, Tanvir Ahmed, and Md Sabbir Ejaz. Weighted score level fusion of iris and face to identify an individual. In *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–4. IEEE, 2017.

[32] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[33] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[34] Dongmei Pei, Chengpu Zhang, Yu Quan, and Qiyong Guo. Identification of potential type ii diabetes in a chinese population with a sensitive decision tree approach. *Journal of diabetes research*, 2019, 2019.

[35] Anne-Laure Boulesteix, Silke Janitza, Jochen Kruppa, and Inke R König. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6):493–507, 2012.

[36] Zhiwen Yu, Zhongkai Deng, Hau-San Wong, and Lirong Tan. Identifying protein-kinase-specific phosphorylation sites based on the bagging–adaboost ensemble approach. *IEEE transactions on nanobioscience*, 9(2):132–143, 2010.

[37] Mohak Shah, Mario Marchand, and Jacques Corbeil. Feature selection with conjunctions of decision stumps and learning from microarray data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):174–186, 2011.

[38] Huma Naz and Sachin Ahuja. Deep learning approach for diabetes prediction using pima indian dataset. *Journal of Diabetes & Metabolic Disorders*, 19(1):391–403, 2020.

[39] Davide Chicco and Giuseppe Jurman. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making*, 20(1):16, 2020.

[40] Kaladevi AC Ramya Perumal. Early prediction of coronary heart disease from cleveland dataset using machine learning techniques. *International Journal of Advanced Science and Technology*, 29(06):4225 – 4234, May 2020.

[41] Md Nahiduzzaman, Md Julker Nayeem, Md Toukir Ahmed, and Md Shahid Uz Zaman. Prediction of heart disease using multi-layer perceptron neural network and support vector machine. In *2019 4th International Conference on Electrical Information and Communication Technology (EICT)*, pages 1–6. IEEE, 2019.

[42] Animesh Kumar Paul, Pintu Chandra Shill, Md Rafiqul Islam Rabin, and Kazuyuki Murase. Adaptive weighted fuzzy rule-based system for the risk level assessment of heart disease. *Applied Intelligence*, 48(7):1739–1756, 2018.

[43] Oluwarotimi Williams Samuel, Grace Mojisola Asogbon, Arun Kumar Sangaiah, Peng Fang, and Guanglin Li. An integrated decision support system based on ann and fuzzy_ahp for heart failure risk prediction. *Expert Systems with Applications*, 68:163–172, 2017.