Hindawi
Scientific Programming
Volume 2021, Article ID 6621622, 12 pages
https://doi.org/10.1155/2021/6621622

Research Article

An Efficient SMOTE-Based Deep Learning Model for Heart Attack Prediction

Muhammad Waqar,[1] Hassan Dawood ,[2] Hussain Dawood ,[3] Nadeem Majeed,[4] Ameen Banjar,[5] and Riad Alharbey[5]

[1]*Department of Electrical Engineering, National University of Modern Languages, Islamabad 74700, Pakistan*
[2]*Department of Software Engineering, University of Engineering and Technology, Taxila 74700, Pakistan*
[3]*Department of Computer and Network Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah 21589, Saudi Arabia*
[4]*Punjab University College of Information Technology (PUCIT), University of the Punjab, Lahore, Pakistan*
[5]*Department of Information System and Technology, College of Computer Science and Engineering, University of Jeddah, Jeddah 21589, Saudi Arabia*

Correspondence should be addressed to Hussain Dawood; hdaoud@uj.edu.sa

Received 3 November 2020; Revised 23 January 2021; Accepted 6 March 2021; Published 16 March 2021

Academic Editor: Mu-Chen Chen

Cardiac disease treatments are often being subjected to the acquisition and analysis of vast quantity of digital cardiac data. These data can be utilized for various beneficial purposes. These data's utilization becomes more important when we are dealing with critical diseases like a heart attack where patient life is often at stake. Machine learning and deep learning are two famous techniques that are helping in making the raw data useful. Some of the biggest problems that arise from the usage of the aforementioned techniques are massive resource utilization, extensive data preprocessing, need for features engineering, and ensuring reliability in classification results. The proposed research work presents a cost-effective solution to predict heart attack with high accuracy and reliability. It uses a UCI dataset to predict the heart attack via various machine learning algorithms without the involvement of any feature engineering. Moreover, the given dataset has an unequal distribution of positive and negative classes which can reduce performance. The proposed work uses a synthetic minority oversampling technique (SMOTE) to handle given imbalance data. The proposed system discarded the need of feature engineering for the classification of the given dataset. This led to an efficient solution as feature engineering often proves to be a costly process. The results show that among all machine learning algorithms, SMOTE-based artificial neural network when tuned properly outperformed all other models and many existing systems. The high reliability of the proposed system ensures that it can be effectively used in the prediction of the heart attack.

1. Introduction

Machine learning and deep learning aim to filter out unknown relations and patterns residing in data. Furthermore, these patterns are used in the construction of various prediction models. Advancement in technology has contributed to the automation of various functional units across multiple domains. Health care is one domain that is generating a vast quantity of complex interrelated data regarding hospitals, patients, and diseases via various electronic devices. This raw data can be a critical resource, but it needs to be processed properly. These data can be processed to extract useful information present within it. Machine learning and deep learning are two major techniques that can accomplish the aforementioned task.

The World Health Organization (WHO) highlighted that one of the core reasons of death among the world population is cardiovascular disease. It includes problems ranging from abnormalities in heart arteries, veins, and deformity in cardiac muscles. This has resulted in a dire need to detect heart disease accurately.

The application of machine learning and data mining techniques in the health care domain has led to the emergence of a new era of computing. Different data mining techniques have been used extensively to detect heart disease effectively [1]. The main problem of machine learning models is that it often requires feature engineering for its effective implementation which often becomes tiresome task [2]. To cater above problem, deep learning has been used extensively for various classification tasks in the health domain particularly in cardiovascular disease [3]. However, the main problem with deep learning is that it generally requires a large dataset for its learning and consumes a lot of resources.

An effective algorithm was proposed to predict heart attack for a given dataset. The main limitation of existing literature was the need of extensive costly feature engineering in the overall classification process. Furthermore, the particular imbalanced nature of the dataset can hinder the overall performance of the classification algorithm in terms of accuracy and reliability. The proposed algorithm aims to minimize the cost associated with features engineering. It is based on end-to-end learning where preprocessed data are fed for classification without any feature engineering. Furthermore, the imbalance nature of the given dataset is explored, and an efficient approach was proposed to improve the reliability of classification results.

The proposed research focuses primarily on the cardiac dataset which is obtained from UCI data repository. Data are preprocessed, and afterward, various machine learning algorithms are applied. The main contributions of the proposed research work are as follows:

(1) To cater imbalanced nature of the given dataset. SMOTE is applied prior to feeding data to these machine learning models so that the imbalance problem of the given dataset can be resolved.

(2) To identify an appropriate classification algorithm that would classify the given dataset accurately.

(3) No feature engineering is performed. More focus is done on tuning the classification model so that it can produce accurate results. So, an effort has been made to generate a cost-effective solution for the given dataset.

(4) To ensure reliability in classification results, proposed work is being evaluated by all state-of-the-art evaluation matrices. Furthermore, K-fold validation is applied to increase reliability of results.

(5) Results are evaluated and compared with recent existing systems which highlighted that proposed research work has outperformed many existing systems results without performing any feature engineering on the given dataset.

The remaining paper has been organized in the following way: Section 2 highlights the literature review section underlining the recent advancements made in the given domain. Section 3 describes the proposed framework which is being proposed for the classification of the given dataset. Section 4 discusses and analyzes classification results that are obtained as a result of the application of the proposed methodology. Section 5 concludes the findings of the proposed work in an effective way.

2. Literature Review

The advancement in technology has led to the need for innovation in every field. The availability of huge bulk of medical data has led to dire of its effective utilization. Health care data are often critical in nature and very difficult to comprehend when it comes to manual processing. Machine learning and data mining techniques are extensively used to explore medical data for the given purpose. These techniques are proving to be very effective when generating solutions for life-critical disease. Heart disease is one life-threatening disease that has been a major cause of death in developing countries [4]. The main reason for such a high fatality rate is due to unidentified risks and patterns associated with cardiac data. Machine learning algorithms are overcoming above mentioned problem by extracting useful patterns from the given dataset in an effective way. Some useful techniques which are being used extensively are support vector machine (SVM), neural networks, decision tree, naïve Bayes classifier, and regression [5]. Furthermore, associative classification has been used to improve the classification of the heart attack prediction.

The existing research has utilized ensemble methods for improving the accuracy of classification models. [6]. An artificial neural network has also been used for the prediction of a heart attack which can effectively increase the accuracy of prediction [7]. Furthermore, many analytical studies have shown that hybrid systems based on associative mining rules are also being used for the heart attack prediction [8]. In this approach, patterns are extracted after which rules are constructed based on these patterns. Some approaches [9] have used heart rate variability analysis to investigate the nature of cardiac signals and also their relative nature while identifying the normal and abnormal signals. Moreover, in recent times, many IOT-based healthcare systems have been proposed [10] which entailed the usage of cardiac data for the effective monitoring of the patient. Furthermore, many studies [11] have underlined the importance of big data in decision support systems. It underlines the core role of big data and how it can be used for various beneficial purposes. A system is proposed to utilize big data to analyze and process brain signals [12] which further shows the effectiveness of deep learning in the healthcare domain.

Furthermore, genetic algorithms [13] have been used for the prediction of heart attack along with the neural network. These hybrid systems use a global optimization approach of genetic algorithm for the initialization of weights for the neural network. Furthermore, fuzzy systems are used along with machine learning algorithms [14] for the prediction of the heart attack. These systems used a fuzzy rule-based approach with the assistance of a decision tree to predict the heart attack. Moreover, the majority of effort in past has been done to extract relevant features from a given dataset in order to improve the accuracy of the algorithm. A clustering-based approach was proposed [15] to extract the relevant features which have more impact on classification results. To avoid uncertainty in clustering, K-mean clustering and spectral clustering are used for the construction of clusters.

Although various methods have been used for predicting heart attack, only some methods are able to predict heart attack with good accuracy. The majority of these algorithms use a hybrid learning approach to generate predictions. Most existing work has undermined the fact that the UCI dataset is somewhat imbalanced where negative examples are less in number as compared to positive examples. This imbalance is often one of the important reasons for getting low accuracy while performing K-fold validation on the given dataset. The minority class contribution in overall prediction results is being surpassed as compared to positive examples due to the presence of a smaller number of instances in the overall dataset. For that very reason, many of the times, minority class results are not being documented properly due to its nonsignificance. This is not acceptable when you are dealing with a critical medical dataset where each instance has its significance in predicting disease. Furthermore, many existing systems have only been measured performance in terms of accuracy. Accuracy often becomes a misleading evaluation matric while dealing with the imbalanced dataset. Proposed work highlights the given imbalance nature of the UCI dataset. An effective approach has been proposed which would overcome the problems that arise while generating predictions from the given imbalance dataset. Furthermore, more standard evaluation matrices are used for the evaluation of results. In the generation of results, more emphasis has been done on evaluating the performance of minority class contribution on the overall performance of the model. On contrary to most existing literature, no feature engineering has been used for generating predictions. Feature engineering often proves to be a very costly process as it involves manual extraction of significant features from the dataset which becomes a tiresome job [16] when the dataset is huge. Furthermore, getting the relevant features involve a deep understanding of the relevant domain [17] for which classification is to be made. Most of the time, there is a considerable number of resources that are used to undergo feature engineering. The main aim of feature engineering is to increase the accuracy of results which may result in increasing biasness within the model. The unique aspect of the proposed research work is that preprocessed data without any feature engineering have been fed to models and results obtained are still comparable to many state-of-the-art existing systems that have used feature engineering for classification This further highlights the effectiveness of the proposed model as the cost-effective solution has been proposed having high reliability.

3. Proposed Framework

The data are preprocessed for missing values and then normalized using the standard scalar technique. Afterward, SMOTE technique is applied to the given dataset to handle the data imbalance problem residing within the dataset. Furthermore, various machine learning algorithms are applied to the given dataset, and their results are evaluated. The main aim was to identify the algorithm which can classify the given dataset in the best manner. The following section thoroughly discusses the proposed framework and its components in detail.

3.1. Dataset. The dataset which has been used is the UCI heart attack dataset. The UCI dataset consists of 303 records having 76 attributes. Most published research works have used subset of 14 attributes. The main reason for choosing above mentioned attributes is that these attributes were considered most important while predicting heart disease for a particular patient. These attributes are specifically mentioned by UCI repository while publishing dataset for public. All published research work on this dataset have used these 14 attributes as they are the most correlated attributes in regard to output class (see reference [18]). For that very reason, these 14 attributes are selected for proposed research work which is shown in Table 1. Table 1 further highlights the relative meaning of attributes present within dataset. These attributes along with their corresponding values show how a particular attribute can be related to heart attack. Among these 14 attributes, 13 attributes are used in prediction of heart disease. One attribute “Target” serves as output variable whose value determine the presence or absence of heart attack. Data preprocessing is done on given dataset before the application of classification models. The main steps applied for data preprocessing were as follows: the data were normalized between 0 and 1 to increase the performance of a model. Missing values were replaced with mean values for corresponding columns. The output class “target” was transformed from multiclass into binary class where 1 represents presence of heart attack and 0 represents absence of heart attack.

3.2. Imbalance Nature of Dataset. The given dataset contains 164 instances of the positive class (1) and 139 instances of the negative class (0). There is an unequal distribution of classes within the dataset. This unequal distribution is one of the major causes of decreasing accuracy of classification models. The main reason is that most machine learning models cannot learn patterns for both positive and negative classes effectively because of their imbalance numbers in a dataset. Moreover, as minority class, i.e., negative class is less in number, so results generated by this class often become ineffective because of its less number. Most literature studies do not document minority class contribution toward producing overall classification results. One of the key contributions of the proposed work is the imbalanced nature of the given dataset is handled effectively via SMOTE technique. Furthermore, results for the majority and minority classes are documented separately in order to evaluate the performance of each class’s contribution in generating overall prediction results.

3.3. Synthetic Minority over Sampling Technique. SMOTE is a famous approach [19] used for the construction of a classifier for the imbalance dataset. An imbalance dataset consists of unequal distribution of underlying output classes. SMOTE is heavily used in classification problems of imbalance dataset [20]. SMOTE preprocessing technique is considered as one of the most reliable techniques when dealing with the imbalanced dataset. Since its release, numerous variants of SMOTE have been proposed and deployed in order to

Table 1: Dataset description.

Sr no.	Data type	Attributes	Description
1	Numeric	Age	It is described in years
2	Binary	Sex	Gender where 1 represents male while 0 represents female
3	Categorical	Chest pain (CP)	It is type of chest pain having four classes with varying degree with values from (0 to 3)
4	Numeric	Resting blood pressure (testbps)	Measure of blood pressure in mmHg (94 to 200)
5	Numeric	Cholestrol (chol)	Measure of cholesterol in blood (126 to 554)
6	Binary	Fasting blood sugar (fbs)	Measure of sugar in blood with fasting (0, 1)
7	Categorical	Resting ECG (restecg)	Measure of ECG in resting phase having three classes: normal, ST defected, and ventricular hypertrophy (0, 1, and 2)
8	Numeric	Thalach	Maximum heart rate in beats per minute (71 to 202)
9	Binary	Exang	Exercise-induced agina (1 and 0)
10	Decimal	Old peak	ST depression induced by exercise (0 to 4.4)
11	Ordinal	Ca	Number of major vessels colored (0 to 3)
12	Categorical	Thal	Represents thallium stress test results (0 = NA, 1 = fixed defect, 2 = normal, 3 = reversible defect)
13	Categorical	Slope	The slop of ST segment having three classes, upward, flat, and downward slopes with values of 0, 1, and 2
14	Binary	Target	Target attributes having two classes where 1 indicates presence of heart attack 0 indicates absence of heart attack

enhance the existing SMOTE technique in terms of more reliability and adaptability under different situations. SMOTE is considered to be one of the most powerful preprocessing techniques in the machine learning and information mining domain [21]. The aim of SMOTE is to undergo interpolation within data of minority class samples so that their numbers can be increased. This helps in achieving generalization in classification. It is one of the most widely used methods which is applied to cater problems which arise due to the imbalanced number of classification instances in a dataset [20]. In SMOTE, minority class is generally oversampled by generating artificial examples. Feature space of minority class is used for the generation of these examples. On the basis of requirement regarding sampling need, numbers of neighbors are chosen. A line is constructed along with the minority class data points via the usage of these neighbors. SMOTE is a very effective technique while dealing with imbalanced datasets. It tends to equalize the number of majority and minority class instants in training examples.

In the proposed algorithm, imblearn library [22] has been used for the implementation of SMOTE in order to deal with the imbalanced dataset. Figure 1 shows the proposed flow of experimentation where at first dataset is processed to remove null values. Afterward, SMOTE technique is applied to the given dataset so that an equal quantity of positive and negative examples can be obtained. Moreover, at the final stage, various machine learning algorithms are applied on the given dataset and corresponding results are obtained. Stratified K-Fold validation is applied to the given dataset to ensure the reliability of the results.

3.4. Tools and Techniques. This algorithm works Tensor Flow, Keras, and anaconda platform are used as a tool for the implementation of proposed work. Tensor flow is used as a backend library to implement neural networks. It uses a graphical processing unit (GPU) and the central processing unit (CPU) to process and analyze large datasets in order to implement neural networks. It is supported by the vast array of machine learning techniques. Keras is the enhanced version of tensor flow library which is used widely in deep learning due to its simplicity and the Jupyter Notebook is used as a development platform to aid in python-based development.

As the dataset size was relatively small, the required machine learning algorithms are run on the machine which is powered by Intel Core i5-3320M (3rd Gen) processor. It has 4 GB RAM and has 500 GB HDD storage. For fast execution particularly for neural network, Google colab is also used along with local machine.

3.5. Machine Learning Models. After the preprocessing of data, various machine learning algorithms are applied. The basic purpose was to understand the impact of each algorithm on the classification of the given dataset.

3.5.1. K-Nearest Neighbors (KNN). The K-nearest neighbors is a type of supervised machine algorithm which is used to classify label dataset. This algorithm works by extracting neighbors for a particular data point. Furthermore, predictions from these neighbors can be used for generating predictions for an unknown data point's label.

In the proposed research work, scikit learn library is used for the implementation of KNN. The hyperparameter and number of neighbor k is set to 3 after tuning it to the given dataset.

3.5.2. Support Vector Machine (SVM). SVM is a supervised machine learning algorithm just like KNN, but it is more efficient than it in terms of cost and accuracy. In SVM, instead of calculating the distance across each data point, the

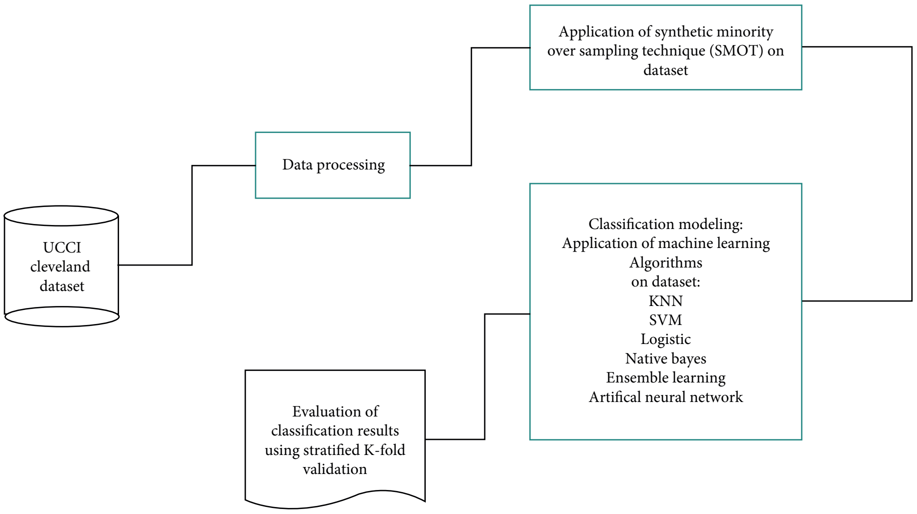


Figure 1: Proposed workflow of experiment.

support vector is calculated across the decision boundary. This support vector is then used for the classification of the given dataset.

In the proposed research work, scikit learn library is used for the implementation of SVM. The "liner" kernel is used for the classification of the given dataset.

3.5.3. Logistic Regression. Logistic regression is another supervised learning algorithm that enables mapping between dependent and independent variables. It is a prediction-based algorithm in which there is a linear combination of variables that predicts a particular output variable.

In the proposed work, scikit learn library is used for the implementation of logistic regression and the corresponding value of output variable "target" is obtained.

3.5.4. Random Forest. Random forest is another supervised learning algorithm that includes combining multiple decision trees. These trees are created using various samples of the training set. Afterward, the prediction is made based on the majority voting of these decision trees. This algorithm is often subjected to overfitting.

In the proposed work, ensemble library from scikit learn package is used for the implementation of random forest hyperparameters are tuned with the number of estimators = 100, max depth = 16, min sample splits = 2, and criteria = gini.

3.5.5. Naïve Bayes Classifier. The naïve Bayes classifier is the supervised classifier of machine learning which works on the Bayes theorem. It is a probability-oriented classifier that works in considering all features of the dataset conditionally independent which means that no correlation exists between features. It is useful for the sparse dataset.

In the proposed work, scikit learn library is used for the implementation of naïve Bayes and the corresponding value of output variable "num" is obtained.

3.5.6. Ensemble Learning. Ensemble learning is a technique in which weaker classifiers are combined, and their results are aggregated for a better and more accurate result. In the proposed work, various ensemble techniques are used in the effort of generating more accurate results for classifying the given dataset.

(i) *Boosting.* Boosting is an algorithm used for implementing ensemble learning. This algorithm works by dividing the dataset into various chunks. The classifier is then trained on these chunks of datasets. Afterward, new subsets of the dataset are created using misclassified data points of the previous iteration. In this way, a more profound model is created for generating more accurate prediction results.

(ii) *Bagging.* Bagging is the famous form of bootstrap aggregation technique. In this approach, each dataset is divided into many subsets. These subsets are selected randomly with replacement. This ensures that each set has the same number of patterns as the old training set. The classifier is trained on these samples. Results are generated using the majority vote. Bagging often results in increase of accuracy for individual weak classifiers.

(iii) *Majority Voting.* It is a type of classifier in which multiple classifiers are combined via meta-models. All classifiers are stacked into layers. Each layer passed its prediction to the next layer. The bottom layer takes input in the form of the dataset and passed it to the upper layer. The topmost layer predicts the final output after aggregating results all layers present below it. The meta-classifier is used to tune the results of stacking models. The final class label would be the class label that had been predicted by a majority of the classifiers.

3.5.7. Neural Network Model. The proposed neural network model is constructed using three layers. One layer is the input layer which contains 12 neurons having "relu" as an activation function. The inner hidden layer, which is used for the construction of the model, contains 8 neurons having "relu" as an activation function. "Relu" function maps all the input to positive values, and if any value is negative, it is changed to 0. The output layer contains "sigmoid" as an output function. This layer differentiates total values obtained from inner hidden layers into two classes, 1 for yes and 0 for no. The architecture is designed to avoid overfitting which greatly reduces the efficiency of the machine learning algorithm. The hyperparameters are tuned with the batch size of 10, epoch value of 150, and the number of the hidden layers is set to one. The hyperparameters are tuned to achieve optimal learning. An effort has been used to reduce the error between the training set and the testing set.

Every neuron gets a lot of x-values (numbered from 1 to n) as an input and process the anticipated $y(i)$ value. Every neuron unit has its features commonly known as w (weight vector) and b (bias) which change during the learning procedure as shown in equation (1). In every iteration, the neuron calculates its output by multiplying "x" with weights and adding bias into the final result. Finally, the result is passed through the nonlinear activation output for the generation of the result as shown in equation (2). These weights are adjusted from time to time after estimating the error difference between actual output and predicted output:

$$\mathbb{Z} = w_1 x_1 + w_2 x_2 + \cdots\cdots + w_n x_n + b. \tag{1}$$

Activation function will be given as

$$\text{output}(n) = \text{activation}(w(n-1) * x) y = g(\mathbb{Z}), \tag{2}$$

where g is the activation function applied on the result obtained from equation (1).

The proposed architecture of the neural network is fully connected which means that each neuron within a layer in connected to all neurons in the other layer. For this reason, more emphasis should be made on how a particular produce an output from a particular input. The general equation for the generation of output for a particular layer can be summarized as follows:

$$\text{output} = \text{activation}(w(n-1) * x). \tag{3}$$

This equation holds for a single layer but for the proposed architecture, one hidden layer is used apart from one input and one output layer. For the second layer, this Output(n) will be considered as input and a similar calculation will be performed by the hidden layer on this Output(n) which is an input for a hidden layer. For the loss function, the binary cross-entropy loss function has been used which is available in keras as shown in equation (4). It is best suited when we have two output binary classes to predict. It is being used in the proposed architecture as we are dealing with binary classification:

$$H(q) = -\frac{1}{N}\sum_{1=1}^{N} Y(i).\ \log(p(y(i)) + (1-y(i)).\ \log(1-p(y(i)), \tag{4}$$

where y is the label (1 for the positive example of heart attack and 0 for the negative example of heart attack) and $p(y)$ is the predicted probability of the point being the positive example of heart attack for all N points. Finally, Adam optimizer [23] has been used for the updation of weights. Adam optimizer is a stochastic gradient-based on the adaptive prediction of 1^{st} order and 2^{nd} order moments. It is computationally efficient, consumes fewer resources, and generally well suited for problems that are large in terms of data and parameters. Figure 2 shows the used neural network model.

One of the important aspects of classification problems in machine learning is about how to tune the hyperparameters for the given classification model. These hyperparameters have a high impact on the overall accuracy and reliability of classification results [24]. Although various methods are defined for hyperparameter tuning [25], yet so far, there is no state-of-the-art algorithm present for hyperparameter tuning. In the proposed research work, grid search technique [26] is deployed while tuning the hyperparameters for different algorithms. In the grid search technique, some random values are given as hyperparameter values. Afterward, the algorithm will find the best value among all possible values for which the best classification results are produced. Table 2 summarizes various values of the hyperparameters which were found out after applying the grid search technique on different algorithms.

4. Result and Discussion

The experiment is performed on the given dataset, and related results are obtained. Stratified K-fold validation is performed for each experiment so that results are free from any biasness. In the proposed research work, no feature engineering is done. The main purpose was to avoid any biasness in the results as feature engineering often results in negligence of some features which may have an impact on overall prediction results. Furthermore, the feature engineering process often proves to be very costly. Raw data after some preprocessing are fed into machine learning algorithms. Afterward, results are obtained and compared with the existing state of the art systems.

4.1. Evaluation Matrices for Imbalance Dataset.

One of the major misconceptions regarding the evaluation of the machine learning model is that every dataset can be measured with the same evaluation matrices regardless of its nature. Most machine learning models tend to be evaluated in terms of accuracy. This approach often proves to be misleading, when we are dealing with the imbalanced dataset [27]. The famous imbalance dataset of fraud detection available on Kaggle [28] is very famous in demonstrating the weakness of the "accuracy" metric for the evaluation of the imbalanced dataset. This dataset has two classes, positive and negative. The positive class comprises of 99% of the dataset and 1% comprises of negative class. If a machine model predicts the positive class for every prediction, then its accuracy will be 99%. This value does not prove to be a true evaluation of a particular machine learning model. Most of the existing literature only focuses on the evaluation of UCI dataset in terms of accuracy. This can be a misleading approach owing to the imbalanced nature of UCI dataset. For that very reason, different standard evaluation matrices are used along

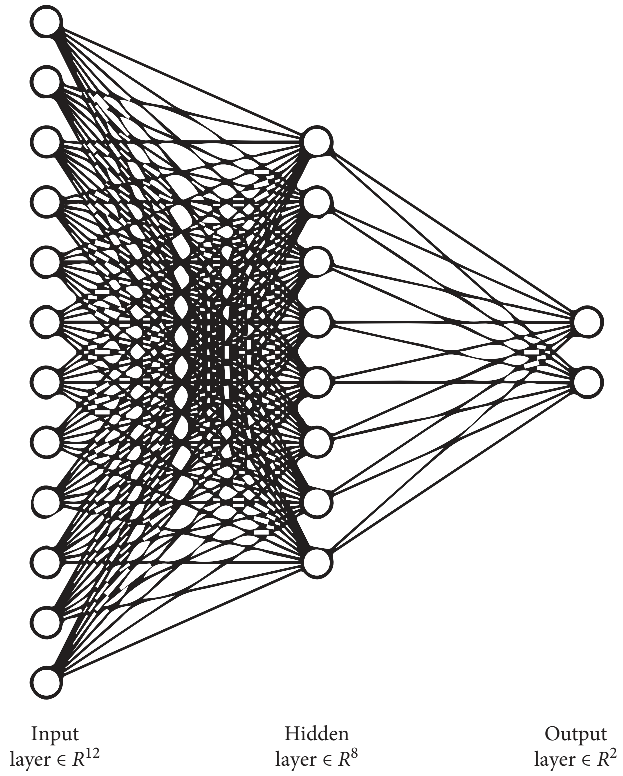


Figure 2: Used neural network model, showing input layer, hidden layer, and output layer.

Table 2: Parameters tunning of classification models.

Classification models	Parameter tunning
AdaBoost	n_estimators = 30, learning rate = 1, algorithm = SAMME.R, base estimator = decisionclassifier
Bagging	Base estimator = decisionclassifier, n_estimators = 100, bootstrap = true
Random forest	n_estimators = 100, max_depth = 16, bootstrap = true, min_samples_leaf = 0.1, min_samples_split = 2, min_leaf_nodes = 10
KNN	n_neighbors = 10, *, weights = “uniform,” algorithm = “auto”, leaf_size = 30
Logistic regression	Stopping criteria-1$e-9$, bias = true, maximum iteration for convergence = 100
Naïve Bayes	Prior = default, smoothing = 1$e-9$, class count = 2
Support vector machine	Regularization = 1, kernel = rbf, kernel coefficient = gamma
Vote	Voting = “hard,” estimators = logistic regression, decision tree,
Artificial neural network	Activation function = “relu” for input layers, activation function = “sigmoid” for output layer, dropout = “0.1 ,” batch size = “10,” num_hidden_layers = “1,” loss = “binary_cross-entropy,” optimizer = “Adam,” epoch = “150”

with accuracy. Precision, recall, $F1$ measure, and ROC curve have been used for the evaluation of the proposed work. Accuracy is the ratio of the number of correct predictions divided by the total number of inputs. The confusion matrices are generated by calculating true positive (TP), true negative (TN), false-positive (FP), and false negative (FN). Sensitivity and specificity are two measures that are calculated as TP/(FN + TP) and FP/(FP + TN), respectively. The receiver operation curve (ROC) is another metric that is widely used to evaluate the classification accuracy of a given model.

A special focus has also been made to evaluate the results of the minority class of the dataset. This is done so that validity of the proposed work can be accurately measured.

4.2. Results of Experimental Work. The experiment is performed with the application of various machine learning algorithms on the given dataset. Table 3 shows the results of various algorithms applied on the given dataset. It shows that the SMOTE-based artificial neural network outperforms all other machine learning techniques in terms of accuracy, precision, recall, F-measure, and ROC value. The SMOTE technique presents machine learning models with data having equal distribution of positive and negative classes. This allows respective models to learn the pattern of minority class, i.e., negative class and apply this learning to unseen test data. The high value of evaluation matrices for the proposed ANN model with SMOTE technique indicates that it has outperformed all other machine learning models used in the experiment.

Table 3: Average accuracy, precision, and $F1$-measure obtained after experimentation.

Algorithm	Computational time in seconds	Average precision	Average recall	Average $F1$-measure	Average accuracy	ROC value
SMOTE-based AdaBoost	61	81.3	80.4	80.2	81	83
SMOTE-based bagging	2	81.7	81	80.6	80	75
SMOTE-based random forest	2	85.1	84.3	84.4	85	82
SMOTE-based KNN	15	82.5	81.8	81.7	82	80
SMOTE-based logistic regression	15	81.8	80.7	80.6	81	70
SMOTE based naïve Bayes	22	81.9	81.5	81.4	82	70
SMOTE-based support vector machine	14	84.5	83	82.9	82	70
SMOTE-based vote	22	82.3	81.5	81	81	69
SMOTE-based artificial neural network	69	96.1	95.7	95.7	96	100

Furthermore, computational time associated with each of the machine learning algorithm is also highlighted in Table 3. Although deep learning models often take remarkable amount of time for their execution, proposed neural network has taken just 69 seconds for its execution. The proposed neural network can be easily run on any machine without much difficulty, other machine learning algorithms may have taken less time for execution; however, their results are not that good. This shows the effectiveness of the neural network model on the given dataset. The most important aspect of proposed research involving ANN along with SMOTE is that no feature engineering has been done on the given dataset. The raw data are fed into all machine learning models, and corresponding results are obtained. This further demonstrates the effectiveness of proposed research work because many existing works of literature put emphasis on feature engineering which ultimately brings more cost in predicting better results. This can reduce the effectiveness of a given prediction model. Our proposed research work works on the whole dataset without any feature-engineered data. Owing to this, the proposed model has still outperformed many of the existing models, without taking too much computational time.

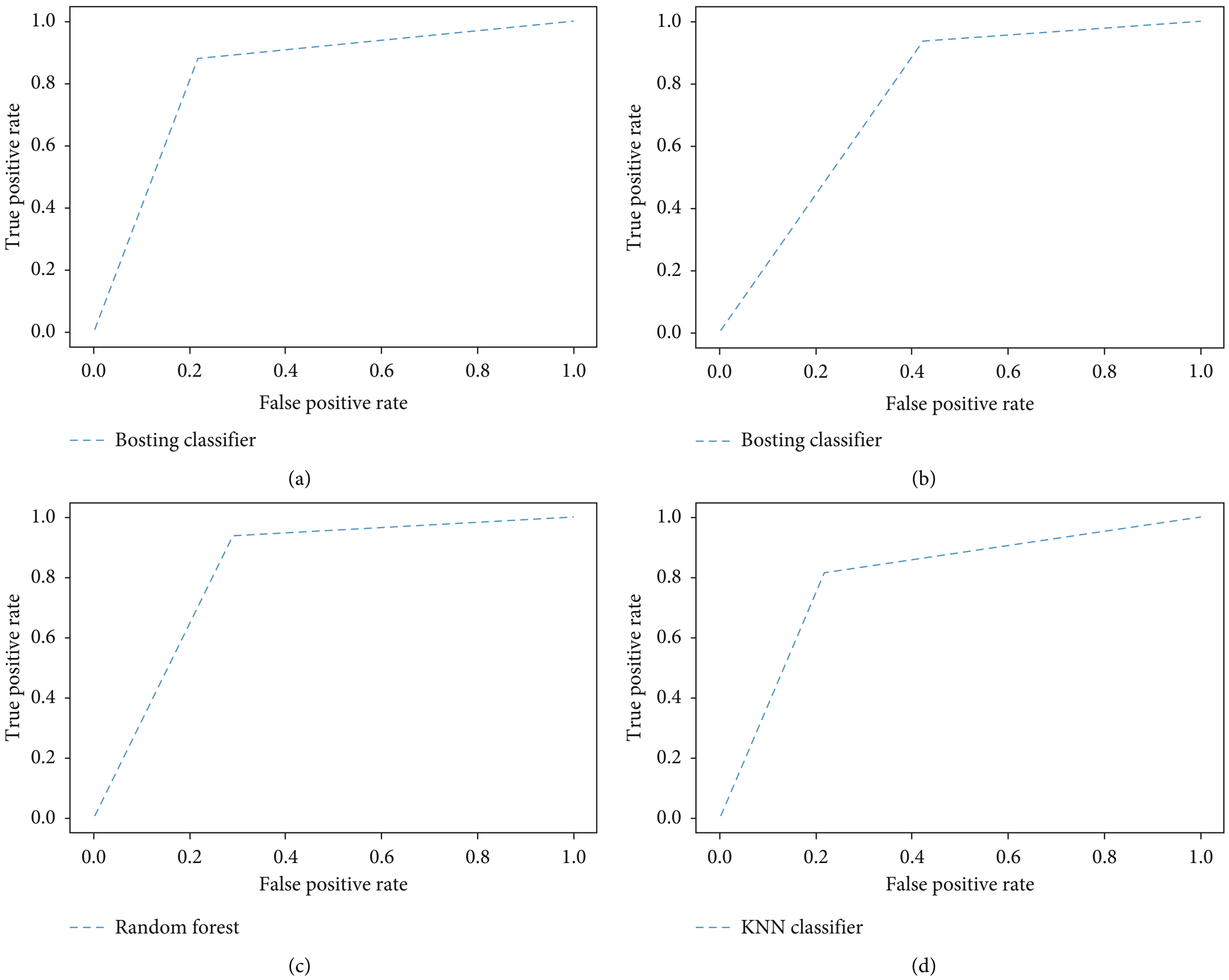
The average accuracy obtained after the application of the proposed model is 96% which has outperformed most state-of-the-art existing systems. The measurement of minority class' impact on classification is very important in dealing with the imbalanced dataset. Machine learning models often ignore minority class contribution while learning patterns for imbalance dataset. Proposed work has specially made emphasis on evaluation of minority class's contribution on the overall accuracy of results generated by machine learning model. Table 4 shows the contribution of each class, i.e., positive and negative class. Results show that the proposed model enables minority class to express its impact on the overall classification process in an effective way. The high values of precision, recall, and $f1$ measure for both positive and negative classes indicate the reliability of the proposed model when measuring results individually for each class as indicated in Table 4. Figure 3 shows ROC curves for 8 various machine learning algorithms used in various algorithms. Figure 4 shows the ROC value of the proposed SMOTE-based neural network has achieved 100% accuracy.

4.3. Comparison with Existing Systems. The results of the proposed work are being compared with results of other state-of-the-art existing systems so that the reliability of the proposed work can be verified. The proposed SMOTE-based artificial neural network model without feature engineering is compared with seven different systems developed in recent years for the given UCI dataset. Table 5 shows the benchmark performance of the proposed model with seven existing systems. The proposed model has outperformed all existing systems. Furthermore, as highlighted earlier, most existing literature only focused on evaluating results in terms of "accuracy" which may become a misleading metric when dealing with the imbalanced dataset. For that very reason, the proposed work has been evaluated with all state-of-the-art evaluation matrices. Table 5 highlights the fact that most existing literature evaluated results on basis of accuracy while for the given framework, accuracy, precision, recall, $F1$-measure, and ROC curve are used for evaluation purposes. It further strengthens the validity of classification results as all state-of-the-art evaluation matrices are used in evaluating results. Results obtained for each metric shows that the proposed system performed exceptionally well in all aspects. Moreover, one more important aspect is that most existing literature has used feature-engineered dataset for the prediction of results which often becomes a costly process. The problem with feature engineering is that it often requires domain knowledge for its execution. Furthermore, sometimes, important information is lost while doing feature engineering. Many times, feature selection is done to increase biasness in results so that accuracy of the classification model can be increased. The proposed model has generated results on the whole dataset without any feature engineering. The results show that our proposed technique has outperformed existing techniques, that so without the involvement of any feature engineering.

4.4. ROC Curve Analysis. Execution of the proposed technique is additionally assessed using trademark (ROC) curve analysis. Figure 3 shows relative ROC values obtained from the various machine learning algorithms when applied in experimentation. ROC curves are plotted against the true

Table 4: Average accuracy, precision, and $F1$-measure obtained for positive and negative classes after experimentation.

Algorithm	Average precision		Average recall		Average $F1$-measure	
	Positive class	Negative class	Positive class	Negative class	Positive class	Negative class
SMOTE-based AdaBoost	81.6	80.6	83.3	77.1	82	78.1
SMOTE-based bagging	81.4	82.2	84.6	76.2	82.4	78.4
SMOTE-based random forest	83.4	86.5	89.6	78.5	86.3	
SMOTE-based KNN	83.9	80.6	82.7	80.6	82.9	80.3
SMOTE-based logistic regression	81.3	82.6	85.3	75.5	82.7	78.1
SMOTE-based naïve Bayes	83.7	79.9	82.9	79.7	83.1	79.6
SMOTE-based support vector machine	82.1	87.3	88.9	76.2	84.7	80.7
SMOTE-based vote	82.1	82.8	85.8	76.3	83.4	78.7
SMOTE-based artificial neural network	97.7	94.1	94.4	97.1	95.9	95.4



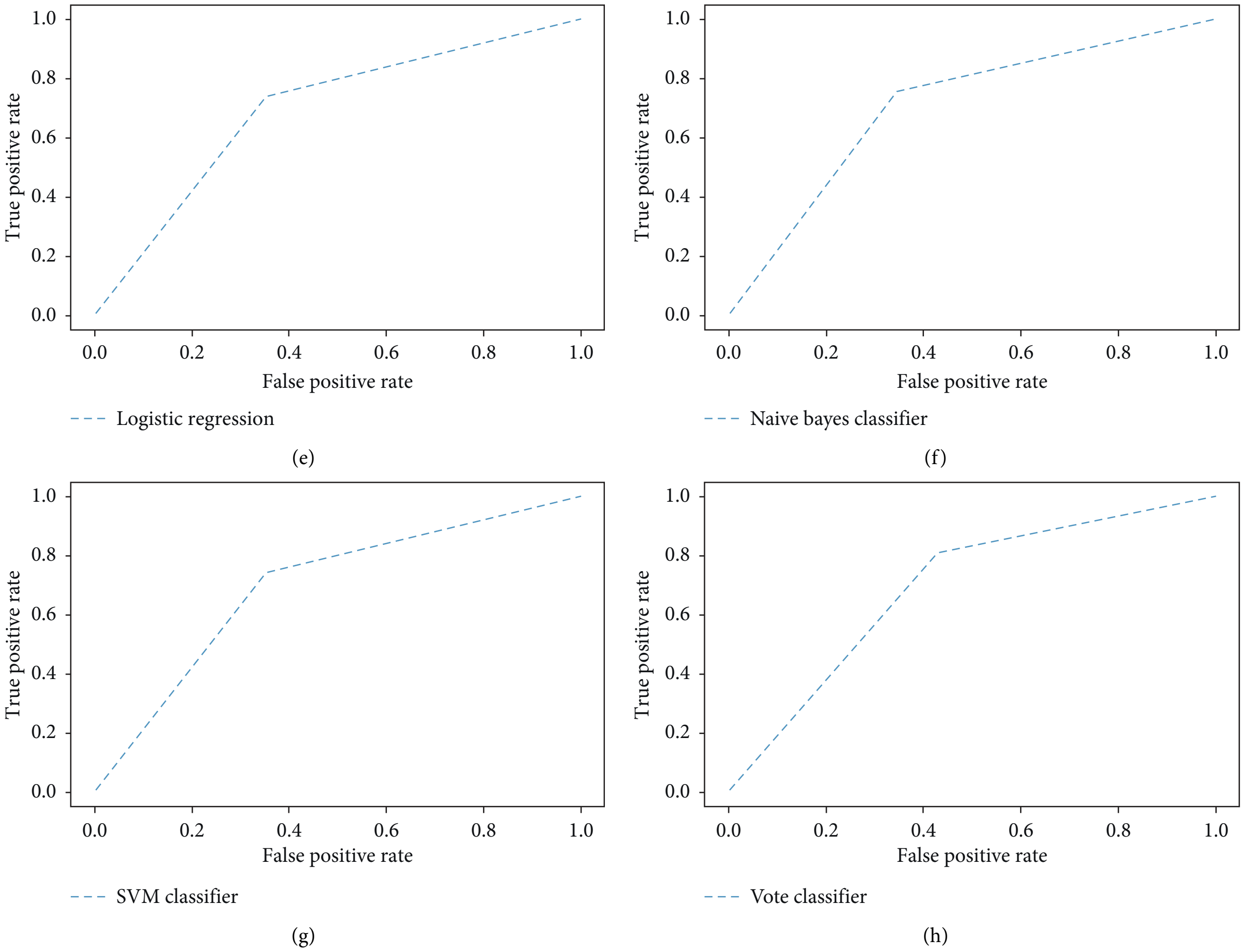
(a)

(b)

(c)

(d)

Figure 3: Continued.



(e) (f) (g) (h)

Figure 3: ROC curves for various algorithms used in the experiment, displaying corresponding values between true positive and false negative.

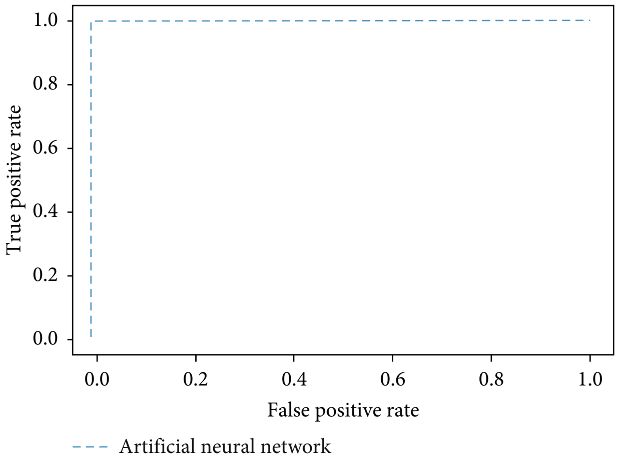


Figure 4: Roc curve for the proposed artificial neural network, showing that proposed algorithm achieved maximum ROC value for the given dataset.

positive rate and false-positive rate. Figure 4 shows the proposed SMOTE-based artificial neural network has achieved a maximum ROC value of 1 when applied to the given dataset. Figure 5 shows the computational time taken by various algorithms. However, deep learning models may consume considerable number of resources for their execution. Contrary to that, the proposed neural network has taken just 69 seconds, and it has produced highly reliable

Table 5: Comparison of existing systems with proposed model.

Source	Technique used	Matrices used for evaluation				
		Accuracy (%)	Precision (%)	Recall (%)	*F*1-measure (%)	ROC curve (AUC) (%)
Shafenoor Amin et al. [6] (with features engineering)	Vote	87.41	79.41		78.10	
C Beulah et al. [29] (with features engineering)	Ensemble learning	85.48				
Raihan et al. [30] (with features engineering)	Artificial neural network	84.47	79.97	82	85.17	
Paul, Shill et al. [31] (with features engineering)	Neural network with fuzzy logic	80				
El-bialey et al. [32] (with features engineering)	Decision tree	78.54				
Subanya et al. [33] (with features engineering)	SVM	86.76				
Djerioui et al. [34] (with features engineering)	Neighborhood component analysis and support vector machines	85.43				
Proposed work (without features engineering)	SMOTE based artificial neural network	96.1	95.7	95.7	96	100

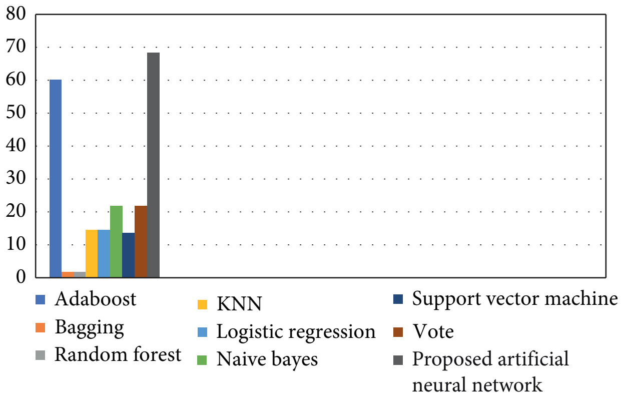


Figure 5: Computational time (in seconds) for various algorithms used in experiment.

results which have outperformed many existing systems. Other machine learning algorithms may have taken less time for execution; however, their results are not that good. This shows the effectiveness of the neural network model on the given dataset.

5. Conclusion

The research work has presented a strategy to predict heart disease from the given dataset. The proposed neural network is one of the best prediction algorithms enables to classify dataset efficiently, without any appreciable data preprocessing. Furthermore, in term of execution time, it has taken remarkably less time but produced highly reliable results. The performance of neural networks depends on the selection of hyperparameters which remain a debatable topic in the construction of the neural network. In the future advance, neural network models such as adversarial neural networks and attention-based neural networks can be applied to the given data set to further improve the accuracy of classification. The similar prediction systems can be developed for other diseases like diabetes or cancer. Moreover, IOT technology may be embedded with the proposed model so that patient's health parameters can be remotely monitored for developing an effective healthcare system.

Data Availability

This work utilizes the standard dataset which is an open access. It can be viewed from the UCI machine learning repository (https://archive.ics.uci.edu/ml/datasets/heart+disease).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] M. S. Amin, Y. K. Chiam, and K. D. Varathan, "Identification of significant features and data mining techniques in predicting heart disease," *Telematics and Informatics*, vol. 36, pp. 82–93, 2019.

[2] H. El-Amir and M. Hamdy, "Feature selection and feature engineering," in *Deep Learning Pipeline*Apress, Berkeley, CA, USA, 2020.

[3] A. Kishore, A. Kumar, K. Singh, M. Punia, and Y. Hambir, "Heart attack prediction using deep learning," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 4, 2018.

[4] World Health Organization, *Assessing National Capacity for the Prevention and Control of Noncommunicable Diseases: Report of the 2019 Global Survey*, World Health Organization, Geneva, Switzerland, 2020.

[5] S. Nalluri, R. Vijaya Saraswathi, S. Ramasubbareddy, K. Govinda, and E. Swetha, "Chronic heart disease prediction using data mining techniques," in *Data Engineering and Communication Technology*, vol. 2020, pp. 903–912, Springer, Singapore, Asia, 2019.

[6] M. Benis Shafenoor Amin, Y. Kia Chiam, and K. Dewi Varathan, "Identification of significant features and data mining techniques in predicting heart disease," *Telematics and Informatics*, vol. 36, 2019.

[7] B. Ballinger, J. Hsieh, A. Singh et al., "DeepHeart: semi-supervised sequence learning for cardiovascular risk prediction," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, LA USA, February 2018.

[8] M. A. Jabbar, P. Chandra, and B. L. Deekshatulu, "Cluster based association rule mining for heart attack prediction," *Journal of Theoretical and Applied Information Technology*, vol. 32, no. 2, pp. 197–201, 2011.

[9] S. Shah, N. Habib, M. Nadeem, A. Alshdadi, M. Alqarni, and W. Aziz, "Extraction of dynamical information and classification of heart rate variability signals using scale based permutation entropy measures," *Traitement du Signal*, vol. 37, no. 3, pp. 355–365, 2020.

[10] A. Rahaman, M. Islam, M. Islam, M. Sadi, and S. Nooruddin, "Developing IoT based smart health monitoring systems: a review," *Revue d'Intelligence Artificielle*, vol. 33, no. 6, pp. 435–440, 2019.

[11] J. Li, M. Nazir Jan, and M. Faisal, "Big data, scientific programming, and its role in internet of industrial things: a decision support system," *Scientific Programming*, vol. 2020, Article ID 8850096, , 2020.

[12] A. Mansoor, M. W. Usman, N. Jamil, and M. A. Naeem, "Deep learning algorithm for brain-computer interface," *Scientific Programming*, vol. 2020, Article ID 5762149, , 2020.

[13] S. U. Amin, K. Agarwal, and R. Beg, "Genetic neural network based data mining in prediction of heart disease using risk factors," in *Proceedings of the IEEE Conference on Information & Communication Technologies*, April 2013.

[14] A. K. Pathak and J. Arul Valan, "A predictive model for heart disease diagnosis using fuzzy logic and decision tree," in *Smart Computing Paradigms: New Progresses and Challenges*, pp. 131–140, Springer, Singapore, Asia, 2020.

[15] J. Bodapati, V. Sajja, N. B. Mundukur, and N. Veeranjaneyulu, "Robust cluster-then-label (RCTL) approach for heart disease prediction," *Ingénierie Des Systèmes D Information*, vol. 24, no. 3, pp. 255–260, 2019.

[16] M. A. Azpúrua, E. Paez, J. Rojas-Mora et al., "A review on the drawbacks and enhancement opportunities of the feature selective validation," *IEEE Transactions on Electromagnetic Compatibility*, vol. 56, no. 4, pp. 800–807, 2014.

[17] P. Przybyszewski, S. Dziewiątkowski, S. Jaszczur, M. Śmiech, and M. Szczuka, "Use of domain knowledge and feature engineering in helping AI to play Hearthstone," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, September 2017.

[18] 2020 http://archive.ics.uci.edu/ml/datasets/Heart+Disease.

[19] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[20] R. Blagus and L. Lusa, "SMOTE for high-dimensional class-imbalanced data," *BMC Bioinformatics*, vol. 14, p. 106, 2013.

[21] A. Fernández, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863–905, 2018.

[22] 2020 https://imbalancedlearn.readthedocs.io/en/stable/generated/imblearn.over_sampling.SMOTE.html.

[23] D. P. Kingma and B. Jimmy, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[24] X. Du, H. Xu, and F. Zhu, "Understanding the effect of hyperparameter optimization on machine learning models for structure design problems," 2020, https://arxiv.org/ftp/arxiv/papers/2007/2007.04431.

[25] M. Feurer and H. Frank, "Hyperparameter optimization," in *Automated Machine Learning*, pp. 3–33, Springer, Cham, Switzerland, 2019.

[26] 2021 https://scikit-learn.org/stable/modules/grid_search.html.

[27] W. Zheng and M. Jin, "The effects of class imbalance and training data size on classifier learning: an empirical study," *SN Computer Science*, vol. 1, no. 2, pp. 1–13, 2020.

[28] 2020 https://www.kaggle.com/mlgulb/creditcardfraud.

[29] C. B. C. Latha and S. C. Jeeva, "Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques," *Informatics in Medicine Unlocked*, vol. 16, p. 100203, 2019.

[30] M. Raihan, P. K. Mandal, M. M. Islam et al., "Risk prediction of ischemic heart disease using artificial neural network," in *Proceedings of the 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, February 2019.

[31] A. K. Paul, P. C. Shill, M. R. I. Rabin, and M. A. H. Akhand, "Genetic algorithm based fuzzy decision support system for the diagnosis of heart disease," in *Proceedings of the 5th International Conference on Informatics, Electronics and Vision (ICIEV)*, pp. 145–150, IEEE, Dhaka, Bangladesh, May 2016.

[32] R. EIBialy, M. A. Salamay, O. H. Karam, and M. E. Khalifa, "Feature analysis of coronary artery heart disease data sets," *Procedia Computer Science*, vol. 65, pp. 459–468, 2015.

[33] B. Subanya and R. R. Rajalaxmi, "Feature selection using artificial Bee colony for cardiovascular disease classification," in *Proceedings of the International Conference on Electronics and Communication Systems (ICECS)*, pp. 1–6, Coimbatore, Tamil Nadu, February 2014.

[34] M. Djerioui, Y. Brik, M. Ladjal, and B. Attallah, "Neighborhood component analysis and support vector machines for heart disease prediction," *Ingénierie Des Systèmes D Information*, vol. 24, no. 6, pp. 591–595, 2019.