Machine Learning-Based Heart Disease Prediction: A Study to Evaluate Driver Fitness

Kanike Keerthana*,1, Goutam Kumar Sahoo**,2, Santos Kumar Das *,3 and Poonam Singh *,4
*Department of Electronics and Communication Engineering, National Institute of Technology, Rourkela, India
Email: 1keerthana.kanike112@gmail.com, 2goutamkrsahoo@gmail.com, 3dassk@nitrkl.ac.in, 4psingh@nitrkl.ac.in

***Abstract*—On-time and efficient identification of heart disease plays a crucial role in healthcare, particularly in cardiology. There is an increased risk of automobile accidents caused by drivers with some cardiovascular disorders. Driving with cardiovascular diseases demands a personalized system for the driver's fitness evaluation, which is essential to reduce traffic accidents. This paper proposes a novel method based on machine learning (ML) techniques that aim to evaluate driver's fitness by finding the significant features that improve heart disease prediction accuracy. This work evaluates the performance of the XG Boost machine learning technology along with previously reported other ML techniques, i.e., the Logistic Regression, K-Nearest Neighbour, Support Vector Machines, Naive Bayes, Decision Tree, Random Forest, and XG Boost. Also, a two-layer convolution neural network is used to test the performance evaluation of the proposed model. Further k-fold cross-validation technique is used to evaluate machine learning models to generalize the classifiers. The performances were evaluated using heart disease datasets from the UCI machine learning repository database. The study achieved the highest accuracy of 90.16% using Random Forest, and 88.52% is obtained for both logistic regression and XG Boost algorithm. The performance comparisons are made with the outcomes of the state-of-the-art heart disease prediction of previous studies. In the future, the aim is to use deep learning approaches for lightweight heart disease prediction models in embedded systems for driver assistance.**

***Keywords*—Cardio Vascular Disease (CVD), Traffic Accidents, Healthcare,, Machine Learning, Heart Disease Prediction.**

I. Introduction

Heart disease or cardiovascular disease (CVD) refers to a condition that affects the heart. In this condition, the heart cannot pump the required amount of blood to human body parts for regular functionalities. As per the report in [1], coronary artery disease (CAD) is the most common type of heart disease, and about 2 in 10 deaths from CAD happen in adults less than 65 years old. The world health organization (WHO) reports over 17.5 million humans die globally per year from heart disease and mainly exhibit myocardial infractions, and heart failures [2], [3]. As per the study, low-income and middle-income countries carry 75% of the CVD deaths worldwide [4]. A prediction was made that more than 30 million deaths would occur due to heart disease by 2040 [5]. Cardiovascular disease and traffic accidents or traffic deaths are the most important researches of recent time for advanced driver assistance systems (ADAS). With advances in automobile technology, it is possible to monitor driver status and the driving environment in a vehicle. The convergence of healthcare in automobiles creates an opportunity to study smart cars for the safety of drivers and healthcare [6]. Heart disease can make a driver lose control of a vehicle without warning and can make a driver unfit for a drive, which may lead to traffic accidents [7]. Unwell drivers with ischemic heart disease increase the relative risk of accidents [8]. Fatal traffic accidents are caused by the sudden failure of the driver to control due to cardiac and other illnesses [9]. Traffic deaths due to sudden heart failure are rare and can not be overlooked in order to save precious lives. The risk assessment of professional drivers with heart disease should be monitored to provide proper healthcare. The fitness to drive with cardiovascular disorders is essential to deal with traffic accidents [10].

With the advancement of artificial intelligence (AI) technology, automatic prediction of heart disease is possible to deal with the healthcare of drivers. Machine learning (ML) has been increasingly used in the medical community, specifically in the domain of cardiovascular diseases [11], which can effectively detect the vital risk factors responsible for predicting heart disease [3]. Using different ML techniques will help in building the best model for heart disease prediction. Importantly models accuracy improvement can be achieved by minimizing the false negatives, i.e., the failure to identify the patients with heart disease [12]. Work presented by Bhatt *et al.* in [13] used the data mining tool Weka to predict heart disease based on the Hungarian dataset and the echocardiogram database [14]. The J48 and Naive Bayes methods show improvement in the range of 82.3% to 98.64% as the accuracy performance metric. However, the Weka tool has an issue in data preparation and visualization. Work by B.S.S. Rathnayakc and G.U. Ganegoda in [15] presented a hybrid method to classify the risk level of a person that uses the K-nearest neighbour (KNN) algorithm, decision trees (DT), genetic algorithm (GA), and Naive Bayes (NB) for heart disease prediction. Also many authors used data mining techniques and artificial neural networks (ANN) for heart disease prediction. Work presented by T. Karaylan and O. Klc in [16] used ANN technique for heart disease prediction using backpropagation algorithm and achieved performance accuracy of 95%. Similarly, the work reported by Olaniyi *et al.* in [17] developed a three-phase technique based on the ANN for heart disease prediction and achieved 88.89% accuracy.

Work presented by Kumar *et al.* in [18] used various machine learning and data mining algorithms trained by the UCI machine learning dataset. The heart disease prediction uses the UCI dataset, which has 303 samples with 14 input features. They found SVM is the best algorithm compared to other algorithms like NB, KNN, and decision tree. Model Performance depends on the variance and the biasness of the datasets. The work presented by H. Sharma and M. Rizvi in [19] compared the performances of ML techniques the

NB and KNN for prediction of heart diseases. NB performs well with low variance and high biasness compared to KNN with high variance and low biasness. Currently, unexpected situations are occurring in diverse environments. The use of data features in various areas like healthcare, traffic accident, etc., can predict future results on different risky situations with the help of ML techniques [20]. Work reported by R. Chitra *et al.* in [21] developed a cardiac disorder prediction system that classifies the risk level of patients based on their medical records and a cascaded neural network (CNN) classifier. The review reported by Mirpuri *et al.* describes the health risks of taxi drivers in different countries with higher rates of CVD compared to non-drivers. The drivers show health exposures to CVD may be due to well-accepted links such as tobacco use, limited physical activity, air pollution, mental health issues, excessive stress, depression, and many more [22]. There is a need to address the CVD risks and conditions of the drivers by predicting heart diseases as an assistance for safe driving.

In this paper, different ML techniques are adopted for the early prediction of CVD to reduce the disease burden. Various performance parameters are used to validate the CVD prediction for individual ML techniques. The best approach may be employed in real-time monitoring of the heart disease severity of drivers to provide driving assistance. Exploratory data analysis (EDA) identifies the critical and the relevant features in the process of modeling. EDA feature analysis in the form of graphical and tabular representation along with different performance metrics will help in deciding the best model for the heart disease classification [23]. The major contributions of this paper on heart disease prediction using machine learning are as follows:

1) Development of a driver assistance model for prediction of CVD to evaluate driver fitness.
2) Improving the accuracy of CVD or heart disease prediction using ML techniques.
3) Performance evaluation of computational intelligence techniques such as LR, KNN, SVM, NB, DT, RF, XG-Boost and CNN for in-vehicle applications.
4) Establishing performance comparison with the state-of-the-art of previous studies on heart disease prediction.
5) Cross validating the performances of the different ML techniques using K-fold cross-validation.
6) The novelty in improving the performances of different ML techniques, XG boosts classifier performance evaluation and cross-validation for model generalization
7) Future studies and possible advancements for the reduction of the risk of automobile accidents.

This study contributes towards the driving assistance of professional drivers having CVDs. The rest of the article is organized as follows: Section II presents the problem formulation with the proposed methodology. Section III explains the experimental results and analysis. The conclusion of the work is mentioned in Section IV.

II. Problem Formulation and Proposed Methodology

CVD is not only the single most significant cause of death, but it is also one of the work-related diseases [24]. There is also a correlation between CVD and occupational factors like working conditions, extreme overwork, loud noise, traffic congestion, poor and irregular diet, working in a sitting position for an extended period every day, and job stress. Professional drivers (bus, taxi, or truck drivers) are mostly exposed to occupational risk factors. The use of alcohol and cigarette to ease psychological problems like anxiety and depression may have a high tendency towards absenteeism, job turnover, accidents and can worsen blood pressure resulting in a higher risk of experiencing CVD among professional drivers. This study formulates a framework to deal with the risk of developing CVD in drivers by early detecting heart diseases using machine learning techniques. The best machine learning models will evaluate the drivers fitness before driving responsibility for the safety of human lives.

The heart disease prediction methodology is represented in the block diagram as shown in Fig.1. A brief explanation of the whole process is as follows. Initially, the dataset is to be collected from the driver before the driving task. For simulation work, the existing database is used here. Next, data pre-processing is done to clean the data for further use. Then features analysis is made using EDA (Exploratory Data Analysis), where different visualization and conclusion are obtained from the features and data collected. The next work is system modeling using different algorithms; finally, model evaluation is performed using the evaluation methods mentioned above. The system block diagram represents step-wise processes followed in carrying out the heart disease detection and classification model development and evaluation. The dataset used has 303 rows and 14 features, divided into 80-20% data splits, namely train and test split, where 80% of the dataset used for training the model using different machine learning algorithms and 20% test dataset used for model validation.

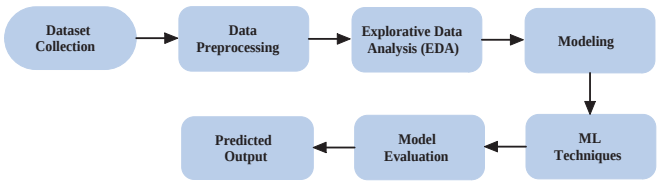


Fig. 1: Heart Disease Prediction Methodology.

A. Dataset Description

For performing heart disease prediction, records of people are primarily essential. This study uses the publicly available Cleveland heart disease dataset from the UCI Machine learning repository [25]. The dataset contains records of 303 individuals with 14 features describing the helpful information for prediction. The details for the dataset are shown in Table I. A detailed view of all the dataset features is shown, which helps in predicting the diseased persons. A brief description of each of the features is given for a better understanding. There are 14 features, of which 13 are considered independent variables/features, and the one, namely the ”Target” feature, is known as the dependent variable/feature. A relationship between an independent and dependent variable will express

TABLE I: Complete Attribute Documentation for Performing Heart Disease Prediction.

Sl. No.	Features	Description
1	Age	Patients age in completed years.
2	Sex	Patients Gender (male = 1 ; female = 0).
3	Cp	The type of Chest pain categorized into 4 values (Typical angina = 1; Atypical angina = 2; Non-angial pain = 3; asymptomatic = 4).
4	Trestbps	Level of blood pressure at resting mode (in mm Hg on admission to the hospital).
5	Chol	Serum cholesterol im mg/dl.
6	Fbs	Blood sugar levels on fasting>120 mg/dl (1 = true; 0 = false).
7	Restecg	Results of an electrocardiogram while at rest are represented in 3 distinct values (Value 0: normal; Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of >0.05 mV); Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria).
8	Thalach	The accomplishment of the maximum heart rate.
9	Exang	Angina induced by exercise (1 = yes; 0 = no).
10	OldPeak	Exercise-induced ST depression in comparison with the state of rest.
11	Slope	ST segment measured in terms of the slope suring peak exercise depicted in three values (Unslopin = 1 ; Flat = 2 ; downsloping = 3).
12	Ca	Fluoroscopy coloured major vessels numbered from 0 to 3.
13	Thal	Status of the heart illustrated through three distinctly numbered values (Normal = 3; fixed defect = 6 ; reversible defect = 7).
14	Target	Heart Disease diagnosis is represented in 5 values (value 0: indicating total absence; Value 1 to 4: representing the presence in different degrees).

the effect of the output variable (dependent variable) on Independent variables.

B. Data Pre-processing

In the real world, data that are not always complete may have missing and noisy data; however, it is always true in the case of medical data. Data pre-processing is used to remove the number of inconsistencies associated with data, duplicate records, normalize the values, account for missing data, etc. The primary step in this data pre-processing is to check null values and treat them by filling or dropping them. After importing the dataset using python library pandas, the data cleaning, data transformation, efficient processing, and classification are carried out. The transformation process changes the data format from one form to another to involve smoothing, normalization, and aggregation tasks [21], [26]. A heatmap is plotted as in Fig. 2 based on the correlation of the data, which gives the relation among different features of the dataset. The figure represents the percentage of correlation for each continuous variable with every continuous variable. A negative value means the variables are negatively correlated, meaning data in one variable is inversely proportional to data in another. Similarly, positive values indicate the correlation is positively related, i.e., data is directly proportional to others. From this plot, we can get the information that which two variables are dependent on each other positively or negatively.

C. Explorative Data Analysis

Exploratory data analysis (EDA) provides the detail features of the dataset through different graphs and tables. From the plot in Fig. 3 shows the relation of a categorical variable with four different categories namely "asymptotic", "non-angial pain", "atypical angina", "typical angina". This plot mainly represents the count of people with these "cp" divisions according to target values. We can observe that people suffering from heart disease are mostly from the "cp" type of "non-angial pain" and people without heart disease are from "typical angina". Similarly the plot in Fig. 4 describes about the feature "thal" compared with "target" and gives the count of people with that specific type. Generally thal gives the status of heart in different conditions as value till 3 as "Normal", value 6 as "fixed defect"' and value 7 as "reversible

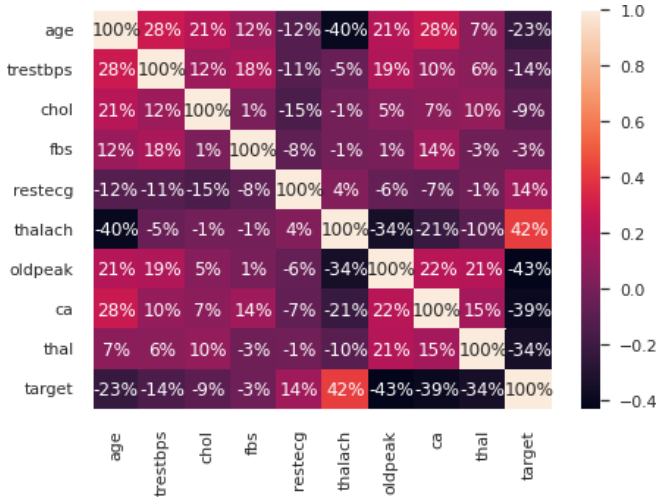


Fig. 2: Heatmap: a correlation of the data.

defect". Our records are from "normal" case which again divided into 4 levels. We can observe that thal value with 2 has more number of people with heart disease. Where as count of people less prone to heart disease with thal value 3 are more in count.

D. Computational Modeling Approaches

Modeling is essential in any prediction problem, where the actual training happens using data mining and machine learning algorithms. A study using a publicly available dataset is made to explore the feasibility of predictive models for the early prediction of heart diseases. A brief description of different machine learning algorithms, the LR, K-NN, SVM, DT, RF, NB, XG Boost, and CNN, are explained in this section.

1) Logistic Regression: Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of the target or dependent variable is dichotomous, where there would be only two possible classes. The learning performances can be visualized using performance parameters like accuracy, F1-scores, etc.

2) K-Nearest Neighbour: K-Nearest Neighbour (KNN) technique is one of the most effective classification techniques.

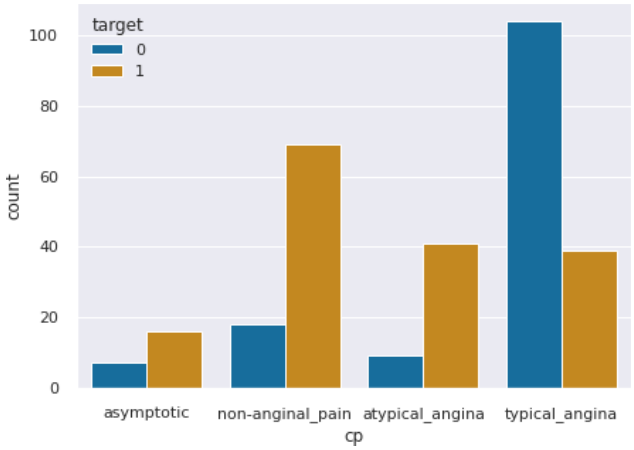


Fig. 3: Distribution of people according to 'cp' Categories.

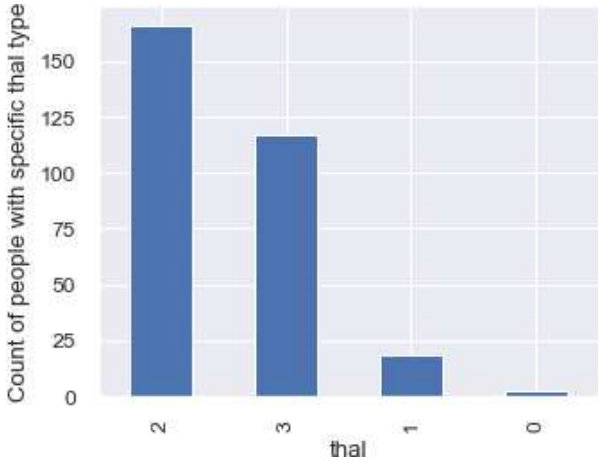


Fig. 4: Distribution of people according to 'thal' Level

It makes no assumptions about the data and is generally used for classification tasks when there is very little or no prior knowledge about the data distribution. This algorithm involves finding the k nearest data points in the training set to the data point for which a target value is unavailable and assigning the average value of the found data points to it.

3) Support Vector Machine: Support Vector Machine is an extremely popular supervised machine learning technique that can be used as a classifier and a predictor. For classification, it finds a hyper-plane in the feature space that differentiates between the classes. An SVM model represents the training data points as points in the feature space, mapped so that points belonging to separate categories are segregated by a margin as wide as possible.

4) Decision Tree: The decision tree is of supervised learning algorithm, performs very well with continuous and categorical attributes. The decision tree algorithm first calculates the entropy of every attribute, and then the dataset is split with the help of the variables or predictors with maximum information gain or minimum entropy.

5) Random Forest: Random Forest is also a popularly supervised machine learning algorithm used for regression and classification tasks but generally performs better in classification tasks. As the name suggests, the Random Forest technique considers multiple decision trees before giving an output. So, it is an ensemble of decision trees. For classification, it uses a voting system and then decides the class, whereas, in regression, it takes the mean of all the outputs of each decision tree.

6) Naive Bayes: Naive Bayes is a simple but effective classification technique that is based on the Bayes Theorem. It assumes independence among predictors, i.e., the attributes or features should not be correlated to one another or should not, in any way, be related to each other. Even if there is dependency, all these features or attributes independently contribute to the probability, which is why it is called Naive.

7) XG Boost: XG Boost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. Artificial neural networks outperform all other algorithms or frameworks in prediction problems involving unstructured data (images, text, etc.). However, when it comes to small-to-medium structured/tabular data, decision tree-based algorithms are considered best-in-class right now.

8) Sequential Model: The sequential model in Keras uses the 'add ()' function to add layers to build the CNN model. The first two layers are Conv2D layers. These are convolution layers that will deal with our input images, which are seen as two-dimensional matrices.

E. K-fold Cross validation

The K-fold cross validation (CV) is a statistical method that estimates the machine learning model's performance accuracy and protects against model overfitting. The data are partitioned into five-fold, seven-fold, and ten-fold cross-validation. The advantage of the cross-validation method is to do observations for both training and test, and each observation is used to test precisely once. The data split of 80-20% is used for training and testing, respectively. The performance measurement parameters: the train and test accuracy, precision, recall, and F1-score are evaluated on selected algorithms to validate the model performance.

F. Evaluation Metrics

After all the modeling process, the main task is to evaluate the model on the test data to check whether a particular machine learning algorithm predicts the persons correctly with and without heart disease. The model can be assessed using evaluation metrics of a python library called scikit-learn. After training the train split dataset (obtained on train-test split) on each of the algorithms mentioned above, take the predictions and evaluate the test split model with the predictions. Using the confusion matrix, we can visualize the performance of computational intelligence techniques. In the confusion matrix, four classification performance indices are present. The definition of these is given below [5]:

TP = True Positive (Correctly Identified)

TN = True Negative (Incorrectly Identified)

FP = False Positive (Correctly Rejected)

FN = False Negative (Incorrectly Rejected)

1) Accuracy: The ratio of correctly classified samples to that of total samples. It is represented as:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)} \quad (1)$$

2) Precision: Precision is calculated as the number of correctly positive predicts divided by the total number of positively predict, given by

$$Precision = \frac{TP}{(TP + FP)} \quad (2)$$

3) Recall (or) Sensitivity: Ratio of correctly classified positive samples to total positive instances, given by

$$Recall = \frac{TP}{(TP + FN)} \quad (3)$$

4) F1 Score: It is the harmonic mean of precision and recall. The range of F1 acore is between 0 and 1. The best value is 1 and the worst value is 0, which can be represented as follows:

$$F1Score = \frac{2TP}{(2TP + FP + FN)} \quad (4)$$

III. Experimental Results and Discussions

A. Experimental Setup

The experiment was implemented in Python 3.8. For appropriate understanding, the result has been run on a single computer (Acer Aspire A515-54G , Intel(R) Core(TM) i5-8265U CPU @1.60GHz, RAM 8 GB) with Windows 10.

B. Results Analysis of ML algorithms

Different machine learning algorithms and neural networks are evaluated using a data split of 80% - 20% for the training and validation process, respectively. Based on the methodologies, results are obtained in terms of evaluation metrics.

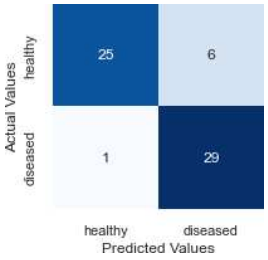
1) Logistic Regression Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 5. From the confusion matrix as in Fig. 5b the algorithm predicts 26 (i.e., 25 correctly classified and 1 wrongly classified), healthy persons, whereas originally, it was 31. The true positive rate (or Recall) is 0.81, i.e., $(25/31)$ and precision is 0.96, i.e., $(25/(25+1))$. Similarly, the diseased class algorithm predicts 35 (i.e., 29 correctly classified and 6 wrongly classified), healthy persons, whereas originally, it was 30. The true positive rate (or Recall) is 0.97, i.e., $(29/30)$ and precision is 0.83, i.e., $(29/(29+6))$. The overall prediction accuracy is found to be 0.89. The learning curve of training data split using logistic regression is shown in Fig. 5c. The learning curve contains two individual graphs, one representing accuracy score on training data and the other representing accuracy score on validation data for continuous training instances. One can see that the maximum training score is 100% and it's gradually decreasing. The minimum value is reached around 87%. But the cross-validation score initially was 80%, then reduced to 75% and reached the maximum of 85%. Therefore, we can conclude that the training score for Logistic Regression gradually decreases, whereas the cross-validation score gradually increases. The training loss graph in Fig. 5d represents the "negative log loss" on training and validation data. Negative Log Loss is defined as a negative average of the log of corrected predicted probabilities for each instance. From the figure, the training loss was initially close to 0, gradually decreasing, and the minimum value is reached around -0.3. But the validation loss score initially was around -0.6, then suddenly increased and finally reached -0.4. Therefore, we can conclude that training loss for logistic regression gradually decreases, whereas cross-validation loss increases.

2) K-Nearest Neighbour Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 6. From the confusion matrix as in Fig. 6b the algorithm predicts 29 (i.e., 25 correctly classified and 4 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.81, i.e., $(25/31)$ and precision is 0.86, i.e., $(25/(25+4))$. Similarly, for the diseased class algorithm predicts 32 (i.e., 26 correctly classified and 6 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.87, i.e., $(26/30)$ and precision is 0.81, i.e., $(26/(26+6))$. The overall prediction accuracy is found to be 0.84. The learning curve of training data-split using K-Nearest Neighbour classifier as shown in Fig. 6c. From the curve, It can be seen that the training performance has a maximum training score of 89% and it's gradually decreasing and reached the minimum value of around 87%, then increased to 86%. But the cross-validation score initially was 83%, then reduced to 77% and reached the maximum of 85%. Therefore, we can conclude that the training score for K-Nearest Neighbour is almost constant, whereas the cross-validation score gradually increases. This Fig. 6d represents the learning curve of the training dataset when trained using K-Nearest Neighbour. We can see that initially, the loss was close to -0.5, and it's gradually increasing to the value of around -0.25. But the cross-validation loss score initially was around -0.75, then suddenly decreased and finally reached -1.75. Therefore, we can conclude that training loss for K-Nearest Neighbour gradually increases whereas, validation loss drops.

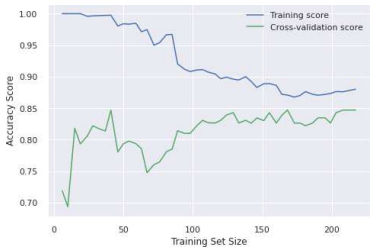
3) Support Vector Machine Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 7. From the confusion matrix as in Fig. 7b the algorithm predicts 23 (i.e., 22 correctly classified and 1 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.71, i.e., $(22/31)$ and precision is 0.96, i.e., $(22/(22+1))$. Similarly, for the diseased class algorithm predicts 38 (i.e., 29 correctly classified and 9 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.97, i.e., $(29/30)$ and precision is 0.76, i.e., $(29/(29+9))$. The overall prediction accuracy is found to be 0.84. The learning curve of training data split using support vector machine is shown in Fig. 7c. The learning curve shows that the maximum training accuracy score is 100%, gradually decreasing, and the minimum value is reached around 89%. But the cross-validation accuracy score initially was 80%, then reduced to 77% and reached the maximum of 83%. Therefore, we can conclude that the

Logistic Regression Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.96	0.81	0.88
Diseased	0.83	0.97	0.89
Macro avg.	0.90	0.89	0.88
Weighted avg.	0.90	0.89	0.89
Accuracy			0.89

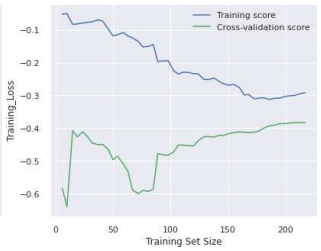
(a) Classification Report



(b) Confusion Matrix



(c) Training Accuracy

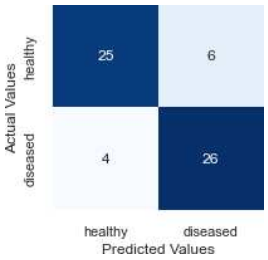


(d) Training Loss

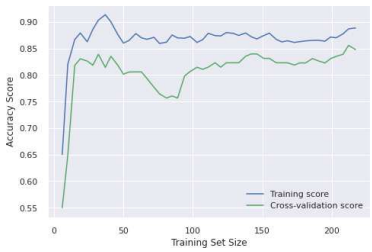
Fig. 5: Logistic-Regression Model Performance.

K-Nearest Neighbour Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.86	0.81	0.83
Diseased	0.81	0.87	0.84
Macro avg.	0.84	0.84	0.84
Weighted avg.	0.84	0.84	0.84
Accuracy			0.84

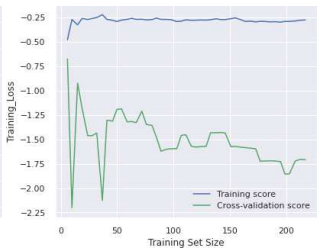
(a) Classification Report



(b) Confusion Matrix



(c) Training Accuracy

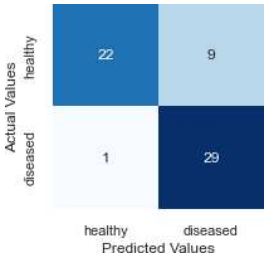


(d) Training Loss

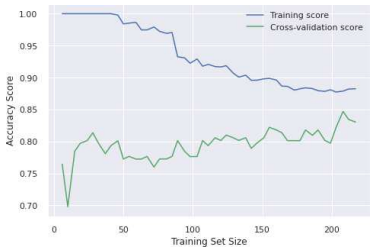
Fig. 6: K-Nearest Neighbour Model Performance.

Support Vector Machine Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.96	0.71	0.81
Diseased	0.76	0.97	0.85
Macro avg.	0.86	0.84	0.83
Weighted avg.	0.86	0.84	0.83
Accuracy			0.84

(a) Classification Report



(b) Confusion Matrix



(c) Training Accuracy

(d) Training Loss

Fig. 7: Support Vector Machine Model Performance.

training score for Support Vector Machines steeply decreases as the cross-validation score gradually increases. This Fig. 7d represents the learning curve of the training dataset when trained using SVM Classifier. Initially, the training loss was close to -0.35, and it's suddenly increased but finally same value as before. But the cross-validation loss score initially was around -0.65, then suddenly increased and finally reached -0.4. Therefore, we can conclude that training loss for SVM Classifier almost remains the same, whereas cross-validation loss gradually increases.

4) Decision Tree Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 8. From the confusion matrix as in Fig. 8b the algorithm predicts 30 (i.e., 24 correctly classified and 6 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.77, i.e., $(24/31)$ and precision is 0.80, i.e., $(24/(24+6))$. Similarly, for the diseased class algorithm predicts 31 (i.e., 24 correctly classified and 7 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.80, i.e., $(24/30)$ and precision is 0.77, i.e., $(24/(24+7))$. The overall prediction accuracy is found to be 0.79. The learning curve of the training data split using decision tree classifier as shown in Fig. 8c. The learning curve shows that the maximum training score is 100%, constant throughout the training process. But the cross-validation score initially was 67%, then gradually increased and reached the maximum of 74%. Therefore, we can conclude that the training score for the decision tree classifier is constant throughout the training, whereas the cross-validation score gradually increases. This Fig. 8d represents the learning curve of the training dataset when trained using Decision Tree Classifier. Initially, the training loss was close to 0 and constant throughout training; however, the cross-validation loss score initially was around -9 then had several zigzag movements and finally remained the same as before. Thus, training loss for Decision Tree

Classifier is constant, whereas cross-validation loss is the same as initial.

5) Random Forest Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 9. From the confusion matrix as in Fig. 9b the algorithm predicts 25 (i.e., 25 correctly classified and 0 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.81, i.e., $(25/31)$ and precision is 1.00, i.e., $(25/(25+0))$. Similarly, for the diseased class algorithm predicts 36 (i.e., 30 correctly classified and 6 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 1.00, i.e., $(30/30)$ and precision is 0.83, i.e., $(30/(30+6))$. The overall prediction accuracy is found to be 0.90. The learning curve of the training data split is shown in Fig.9c. The learning curve shows that the maximum training score is 100%, constant throughout the training process. But the cross-validation score initially was 83%, then decreased to 77% and reached the maximum of 82%. Therefore, we can conclude that the training score for Random Forest Classifier is constant throughout the training, whereas the cross-validation score is finally below than starting score. This Fig. 9d represents the learning curve of training loss which was initially close to -0.11 and constant throughout training. The cross-validation loss score initially was around -0.56, then increased and finally reached -0.41. Hence, we can conclude that training loss for Random Forest Classifier is constant, whereas cross-validation loss gradually increases.

6) Naive Bayes Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 10. From the confusion matrix as in Fig. 10b the algorithm predicts 18 (i.e., 17 correctly classified and 1 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.55, i.e., $(17/31)$ and precision is 0.94, i.e., $(17/(17+1))$. Similarly, for the diseased class algorithm predicts 43 (i.e., 29 correctly classified and 14 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.97, i.e., $(29/30)$ and precision is 0.67, i.e., $(29/(29+14))$. The overall prediction accuracy is found to be 0.75. The learning curve of training data split using Naive Bayes as shown in Fig. 10c. The learning curve shows that the maximum training score is 98%, and it's suddenly decreased to 70% and then gradually increased to 78%. But the cross-validation score initially was 72%, then reduced to 64% and reached the maximum of 75%. Therefore, we can conclude that the training score for Naive Bayes steeply drops, whereas cross-validation remains almost the same as before. This Fig. 10d represents the learning curve of the training dataset when trained using Gaussian NB Classifier. We can observe; initially, the loss was close to 0, and it's suddenly decreased and finally reached around -2. But the cross validation score initially was around -10, then suddenly increased and finally reached -2.5. Therefore, we can conclude that training loss for Gaussian NB Classifier gradually decreases, whereas cross validation loss gradually increases.

7) XG Boost Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 11. From the confusion matrix as in Fig. 11b the algorithm predicts 28 (i.e., 24 correctly classified and 4 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.77, i.e., $(24/31)$ and precision is 0.86, i.e., $(24/(24+4))$. Similarly, for the diseased class algorithm predicts 33 (i.e., 26 correctly classified and 7 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.87, i.e., $(26/30)$ and precision is 0.79, i.e., $(26/(26+7))$. The overall prediction accuracy is found to be 0.82. The learning curve of the training data split using XG Boost classifier as shown in Fig.**??**. The learning curve shows that the initial training score was around 91%, and it's gradually increased to reach the maximum of 100% and finally settled at 98%. But the cross-validation score initially was 76%, and had many zig-zag movements and finally reached 79%. Therefore, we can conclude that the training score for XG Boost gradually increased, whereas the cross-validation score doesn't have a gradual increase but overall increased than initial. This Fig. 11d represents the learning curve of training loss that was initially close to -0.65 and then steeply increases and reached -0.1. The cross-validation loss score initially was around -0.7, then steeply increased and finally reached -0.5. Hence it concludes that training loss for XG Boost Classifier increases steeply, whereas cross-validation loss gradually increases.

8) Sequential Model Classification Report: The classification report, confusion matrix, and the training accuracy performances can be seen from Fig. 12. From the confusion matrix as in Fig. 12b the algorithm predicts 26 (i.e., 22 correctly classified and 4 wrongly classified) healthy persons whereas originally, it was 31. The true positive rate (or Recall) is 0.71, i.e., $(22/31)$ and precision is 0.85, i.e., $(22/(22+4))$. Similarly, for the diseased class algorithm predicts 35 (i.e., 26 correctly classified and 9 wrongly classified) healthy persons whereas originally, it was 30. The true positive rate (or Recall) is 0.87, i.e., $(26/30)$ and precision is 0.74, i.e., $(26/(26+9))$. The overall prediction accuracy is found to be 0.74. Fig. 12c and Fig. 12d describe the accuracy and loss plots obtained from training performances of the CNN model. The training was performed with a mini-batch size of 4, training and validation data split of 80-20% for 300 epochs with binary cross-entropy loss function and adam optimizer.

C. Result Analysis of K-fold Cross Validation

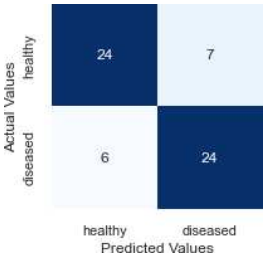
TABLE II: Performance Comparisons in Terms of Test Accuracy using K-fold Cross-Validation

Algorithms	**5 Fold**	**7 Fold**	**10 Fold**
Logistic Regression	83.6	88.52	83.6
Support Vector Machines	83.6	85.24	83.6
Decision Tree	80.32	75.4	78.68
Random Forest	86.88	86.88	86.88

The performance accuracy of the proposed model is evaluated and compared with different k-fold cross-validation techniques. The average accuracy of k-fold cross-validation of the model when trained for five epochs are 87.63%, 88.35%, and 93.05%, for 5-fold, 7-fold, and 10-fold, respectively. A comparative analysis of different model's performance with

Decision Tree Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.80	0.77	0.79
Diseased	0.77	0.80	0.79
Macro avg.	0.79	0.79	0.79
Weighted avg.	0.79	0.79	0.79
Accuracy			0.79

(a) Classification Report



(b) Confusion Matrix

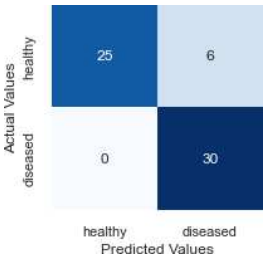
(c) Training Accuracy

(d) Training Loss

Fig. 8: Decision Tree Model Performance.

Random-Forest Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	1.00	0.81	0.89
Diseased	0.83	1.00	0.91
Macro avg.	0.92	0.90	0.90
Weighted avg.	0.92	0.90	0.90
Accuracy			0.90

(a) Classification Report



(b) Confusion Matrix

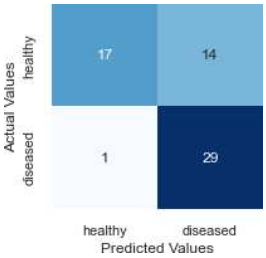
(c) Training Accuracy

(d) Training Loss

Fig. 9: Random Forest Model Performance.

Naive Bayes Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.94	0.55	0.69
Diseased	0.67	0.97	0.79
Macro avg.	0.81	0.76	0.74
Weighted avg.	0.81	0.75	0.74
Accuracy			0.75

(a) Classification Report



(b) Confusion Matrix

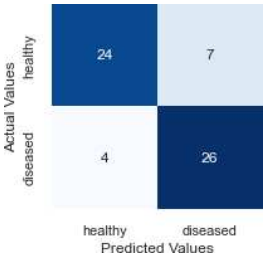
(c) Training Accuracy

(d) Training Loss

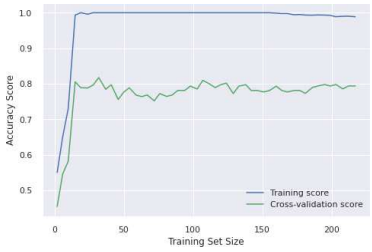
Fig. 10: Naive Bayes Model Performance.

XG Boost Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.93	0.84	0.88
Diseased	0.85	0.93	0.89
Macro avg.	0.89	0.89	0.89
Weighted avg.	0.89	0.89	0.89
Accuracy			0.89

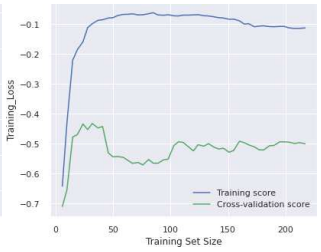
(a) Classification Report



(b) Confusion Matrix



(c) Training Accuracy



(d) Training Loss

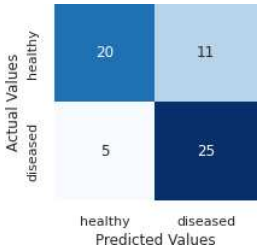
Fig. 11: XG Boost Model Performance.

a k-fold cross-validation approach can be observed from the TABLE II.

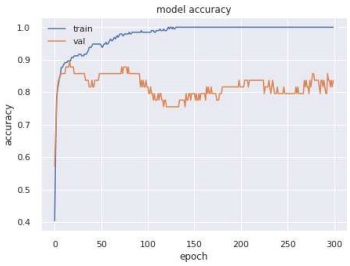
The model accuracy is improved by hyper-parameter tuning using cross-validation techniques. The performances of different cross-validation methods like 5-Fold CV, 7-Fold CV, and 10-Fold CV using grid search CV method on different algorithms are evaluated and which can be observed from the TABLE III, TABLE IV and TABLE V. The TABLE

Sequential Model			
Parameters	**Precision**	**Recall**	**F1 score**
Healthy	0.80	0.65	0.71
Diseased	0.69	0.83	0.76
Macro avg.	0.75	0.74	0.74
Weighted avg.	0.75	0.74	0.74
Accuracy			0.74

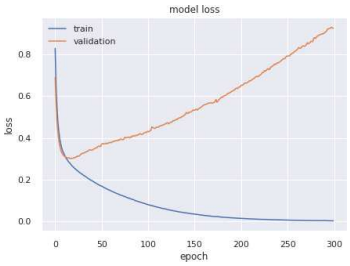
(a) Classification Report



(b) Confusion Matrix



(c) Training Accuracy



(d) Training Loss

Fig. 12: CNN Model Performance.

TABLE III: Performance Evaluation of 5-fold Cross-Validation.

ALGORITHMS	TRAIN_ACCURACY (%)	TEST_ACCURACY (%)	PRECISION (%)	RECALL (%)	F1 SCORE (%)
Logistic Regression	85.95	83.6	83	83	83
Support Vector Machines	87.6	83.6	78	93	85
Decision Tree	85.5	80.32	78	83	81
Random Forest	93.38	86.88	82	93	87

TABLE IV: Performance Evaluation using 7-fold Cross-Validation.

ALGORITHMS	TRAIN_ACCURACY (%)	TEST_ACCURACY (%)	PRECISION (%)	RECALL (%)	F1 SCORE (%)
Logistic Regression	87.19	88.52	83	97	89
Support Vector Machines	87.19	85.24	80	93	86
Decision Tree	83.05	75.4	69	90	78
Random Forest	90.4	86.88	82	93	87

TABLE V: Performance Evaluation using 10-fold Cross-Validation.

ALGORITHMS	TRAIN_ACCURACY (%)	TEST_ACCURACY (%)	PRECISION (%)	RECALL (%)	F1 SCORE (%)
Logistic Regression	85.95	83.6	83	83	83
Support Vector Machines	87.6	83.6	78	93	85
Decision Tree	87.19	78.68	84	70	76
Random Forest	88.4	86.88	82	93	87

TABLE VI: Performance Evaluation of Different Machine Learning Models.

ALGORITHMS	TRAIN_ACCURACY (%)	TEST_ACCURACY (%)	PRECISION (%)	RECALL (%)	F1 SCORE (%)
Logistic Regression	87.19	88.52	82.85	96.66	89.23
K Nearest Neighbour	88.84	83.6	81.25	86.66	83.87
Support Vector Machines	89.25	83.6	76.31	96.66	85.29
Naive Bayes	76.44	75.4	67.44	96.66	79.45
Decision Tree	100	78.68	77.41	80	78.68
Random Forest	**100**	**90.16**	**83.33**	**100**	**90.9**
XG Boost	98.34	88.52	84.84	93.33	88.88
Convolution Neural Network	95.87	78.69	74.29	86.67	80.00

III represents the evaluation outcomes of some selected algorithms, namely Logistic Regression, SVM, Decision Tree, Random Forest using cross-validation. We have performed the 5fold CV on these algorithms to improve the model accuracy. But we can observe that the test accuracy of the decision tree model has improved, and the rest have low value. This motivates us to perform the 7fold CV on the same algorithms to get better performance results. Table IV gives the evaluation outcomes of 7-fold CV algorithms based on Logistic Regression, SVM, Decision Tree, Random Forest. The objective was to improve all model accuracy; however, we can observe that test accuracy of SVM, Decision Tree, and Random Forest models improved somehow, but the rest don't have any effect. Further, we perform the same with 10fold CV on the same algorithms to get the best results. Similarly, Table V gives the evaluation outcomes of 10-fold CV to improve the model accuracy; however, we can observe that the test accuracy of SVM, Decision Tree models increased, and the rest do not have much improvement.

D. Comparative Analysis of Heart Disease Prediction

The testing accuracy evaluated and performances are compared in different techniques, which can be seen in Fig. 13. It represents the comparison plot of the testing accuracy

evaluated using different algorithms used for training the model. We can conclude from the bar graph that testing accuracy using the Random forest algorithm reached the highest of 90.16%, and the least accurate model is Naive Bayes. The evaluation metrics are calculated, and their representation can be visualized from Table VI. The metrics considered in this are Train accuracy, Test accuracy, Precision, Recall, and F1 Score, which are the most important metrics to decide the better model. From the table, we can conclude that Random Forest has a maximum test accuracy of 90.16%.

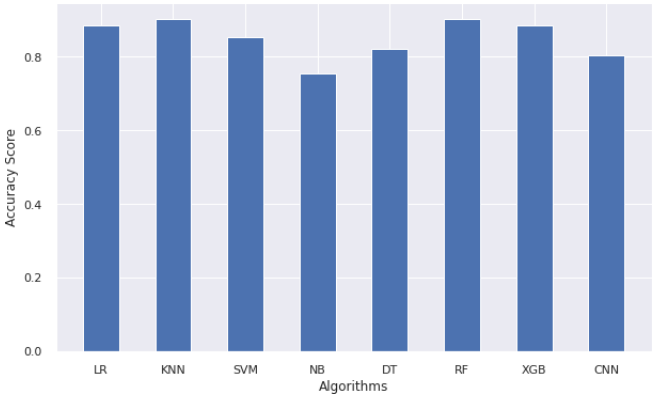


Fig. 13: Accuracy Comparison of Different Models.

In Table VII, the accuracy result shows that our experimental study on heart disease prediction using Logistic Regression, K-Nearest Neighbour, Support Vector Machines, Naive Bayes, Decision Tree, Random Forest, and XG Boost. The performances of logistic regression technique applied in [30], [32]–[34], [37], [39] achieved the accuracy performances of 82.8%, 85%, 87.8%, 83.5%, 83.3%, and 73.77% respectively, which are somehow similar and lower than the performance of our study. The K-nearest neighbor technique was used in [32], [36], [37], [39] and the performances achieved 80%, 71.42%, 84.8%, and 75.4% respectively; however, the accuracy performances are similar and lower than our study. The work presented in [27], [29] used Support Vector Machines(SVM) and achieved an accuracy of 84.12% and 84.5%, which are similar to our result. Similarly the study presented in [32], [34], [37] used SVM, however performance is lower than our proposed study. Pouriyeh et al. [31] presented Boosting with SVM algorithm and achieved an accuracy of 84.51%. The study presented in [5], [26], [37], [38] used Random Forest and achieved the accuracy performance of 86.84%, 89.41%, 80.30% and 89.97% respectively. However, our study improves the accuracy performances to 90.16%. G. Parthiban and S. K. Srivatsa [28] applied Naive Bayes classifier using the WEKA tool to predict the heart diseases in diabetic patients, and the model was able to classify 74% of the input correctly, whereas our study performs better here. The XG Boost algorithm performance is evaluated on the same dataset, and good accuracy performance is achieved, which is 88.52%; however, the implementation of the XG Boost algorithm is not considered by authors presented in the comparative study as in Table VII. The work presented by Ayon *et al.* [5] shows performance improvement on all algorithms except the Random Forest, where our study performs better. Similarly, work presented by R. Katarya and S. K. Meena [36] shows the low performance when evaluated with the K-nearest neighbor technique. Our proposed study provides accuracy improvement on most of the methods as discussed. The maximum accuracy of 90.16% is achieved in Random Forest and 88.52% in both logistic regression and XG Boost algorithm. Some authors also present the work of the deep learning-based approach. Our study with deep neural networks like convolution neural networks (CNN) achieves an accuracy of 73.77%, which can be further improved with the help of existing lightweight deep learning models in the near future.

TABLE VII: Comparison of Our Study With Existing Methods

Year	Author	Methods	Accuracy (%)
2011	M. Kumari and S. Godara [27]	SVM	84.12
2012	G. Parthiban and S. K. Srivatsa [28]	Nave Bayes	74
2015	Otoom et al. [29]	Naive Bayes SVM Functional Trees	84.5
2015	Khanna et al. [30]	Logistic Regression Neural Network	82.8 89
2017	Pouriyeh et al. [31]	Boosting with SVM	84.51
2018	A. K. Dwivedi [32]	Naive Bayes Decision Tree KNN Logistic Regression SVM ANN	83 77 80 85 82 84
2020	Shah et al. [26]	Naive Bayes KNN Decision Tree Random Forest	88.15 90.78 80.26 86.84
2020	M. A. Khan [33]	Deep Neural Network Logistic Regression	81.8 87.8
2020	Ayon et al. [5]	SVM Logistic Regression Deep Neural Network Decision Tree Naive Bayes Random Forest KNN	97.36 92.41 94.39 92.76 91.18 89.41 94.28
2020	Tougui et al. [34]	Logistic Regression SVM	83.5 83.16
2021	sapra et al. [35]	Ensemble learning	84
2021	R. Katarya and S. K. Meena [36]	Logistic Regression Naive Bayes SVM KNN Decision Tree Random Forest ANN DNN MLP	93.4 90.1 92.3 71.42 81.31 95.6 92.3 76.92 75.42
2021	Bharti et al. [37]	Logistic regression K neighbors SVM Random forest Decision tree DL	83.3 84.8 83.2 80.3 82.3 94.2
2021	Divya et al. [38]	Support vector machine Decision tree (CART) Random forest Naive Bayes Logistic regression	87.995 88.004 89.965 89.057 92.009
2021	Md R. Rana and N. Al-Musabbir [39]	Logistic Regression K-Nearest Neighbors	73.77 75.409
Our Study		**Logistic Regression** **K Nearest Neighbour** **Support Vector Machines** **Naive Bayes** **Decision Tree** **Random Forest** **XG Boost** **CNN**	**88.52** **83.6** **83.6** **75.4** **78.68** **90.16** **88.52** **73.77**

IV. Conclusions and Future Works

This paper presents a novel method for driver fitness evaluation on heart disease prediction using ML techniques. This approach also verifies the performance of the XG Boost machine learning technique for heart disease prediction. The most effective and well-known method, the K-fold

cross-validation, is used to generalize the model performances. The objective is to check the status of driver fitness as healthy or diseased using ML techniques based on specific evaluation metrics. Compared with all the machine learning algorithms, the ensemble model, i.e., Random Forest, gave the maximum test accuracy score of 90.16%. Also, the FP (False Positives) in the confusion matrix is 0, which means the model exactly predicts the person with heart disease as diseased rather than healthy. So, on the whole, Random Forest is the best model. The performances of the models are also cross-validated using k-fold cross-validation techniques. The above predictions are essential to notify the doctor before the seriousness of the disease and to start the medication. In the future, deep learning approaches can be experimented with to improve the prediction accuracy with an objective for applications in embedded systems.

REFERENCES

[1] E. J. Benjamin, *et al.*, "Heart disease and stroke statistics2019 update: a report from the american heart association," *Circulation*, vol. 139, no. 10, pp. e56–e528, 2019.

[2] M. Sanz *et al.*, "Periodontitis and cardiovascular diseases: Consensus report," *Journal of Clinical Periodontology*, vol. 47, no. 3, pp. 268–288, 2020.

[3] D. Chicco and G. Jurman, "Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone," *BMC medical informatics and decision making*, vol. 20, no. 1, pp. 1–16, 2020.

[4] Y. Zhao *et al.*, "Social determinants in machine learning cardiovascular disease prediction models: A systematic review," *American journal of preventive medicine*, 2021.

[5] S. I. Ayon, M. M. Islam, and M. R. Hossain, "Coronary artery heart disease prediction: a comparative study of computational intelligence techniques," *IETE Journal of Research*, pp. 1–20, 2020.

[6] D.-H. Shin, K. Chung, and R. C. Park, "Prediction of traffic congestion based on lstm through correction of missing temporal and spatial data," *IEEE Access*, vol. 8, pp. 150 784–150 796, 2020.

[7] H. H. Klein, U. Sechtem, and H.-J. Trappe, "Fitness to drive in cardiovascular disease," *Deutsches Ärzteblatt International*, vol. 114, no. 41, p. 692, 2017.

[8] A. G. Lledó, E. V. Rodríguez, and M. O. Val, "Heart disease and vehicle driving: novelties in european and spanish law," 2018.

[9] M. Halinen and A. Jaussi, "Fatal road accidents caused by sudden death of the driver in finland and vaud, switzerland," *European heart journal*, vol. 15, no. 7, pp. 888–894, 1994.

[10] F. J. Álvarez *et al.*, "Assessment of fitness to drive and cardiovascular diseases at the spanish medical traffic centres," *Circulation journal*, vol. 71, no. 11, pp. 1800–1804, 2007.

[11] S. J. AlAref *et al.*, "Clinical applications of machine learning in cardiovascular disease and its relevance to cardiac imaging," *European heart journal*, vol. 40, no. 24, pp. 1975–1986, 2019.

[12] R. A. Haraty, M. Dimishkieh, and M. Masud, "An enhanced k-means clustering algorithm for pattern discovery in healthcare data," *Int. Journal Dist. Sensor Networks*, vol. 11, no. 6, p. 615740, 2015.

[13] A. Bhatt, S. K. Dubey, A. K. Bhatt, and M. Joshi, "Data mining approach to predict and analyze the cardiovascular disease," in *Proc. 5th Int. Conf. Frontiers in Intelli. Comput.: Theory and Applications*. Springer, 2017, pp. 117–126.

[14] D. Dua and C. Graff, "UCI Machine Learning Repository, University of California, School of Information and Computer Science, Irvine, CA, 2019."

[15] B. S. S. Rathnayakc and G. U. Ganegoda, "Heart diseases prediction with data mining and neural network techniques," in *Proc. IEEE 3rd Int. Conf. Converg. Techno. (I2CT)*, 2018, pp. 1–6.

[16] T. Karayılan and Ö. Kılıç, "Prediction of heart disease using neural network," in *Proc. IEEE Int. Conf. Comput. Science Engg. (UBMK)*, 2017, pp. 719–723.

[17] E. O. Olaniyi, O. K. Oyedotun, and K. Adnan, "Heart diseases diagnosis using neural networks arbitration," *Int. Journal Intelli. Sys. Applications*, vol. 7, no. 12, p. 72, 2015.

[18] M. N. Kumar, K. Koushik, and K. Deepak, "Prediction of heart diseases using data mining and machine learning algorithms and tools," *Int. Journal Scient. Res. Comput. Science, Engg. Inform. Techno.*, vol. 3, no. 3, pp. 887–898, 2018.

[19] H. Sharma and M. Rizvi, "Prediction of heart disease using machine learning algorithms: A survey," *Int. Journal Recent Innov. Trends Comput. Commun.*, vol. 5, no. 8, pp. 99–104, 2017.

[20] J.-W. Baek and K. Chung, "Accident risk prediction model based on attention-mechanism lstm using modality convergence in multimodal," *Personal and Ubiquitous Computing*, pp. 1–11, 2021.

[21] R. Chitra and V. Seenivasagam, "Heart disease prediction system using supervised learning classifier," *Bonfring Int. Journal Soft. Engg. Soft Comput.*, vol. 3, no. 1, pp. 01–07, 2013.

[22] S. Mirpuri *et al.*, "Cardiovascular health status of taxi/for-hire vehicle drivers in the united states: A systematic review," *Work*, no. Preprint, pp. 1–18, 2021.

[23] C. C. Aggarwal, *Data mining: the textbook*. Springer, 2015.

[24] S. Y. Shin, C. G. Lee, H. S. Song, S. H. Kim, H. S. Lee, M. S. Jung, and S. K. Yoo, "Cardiovascular disease risk of bus drivers in a city of korea," *Annals Occupat. Environ. Med.*, vol. 25, no. 1, pp. 1–9, 2013.

[25] UCI Machine learning repository., *Heart Disease Dataset.* [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Heart+Disease.

[26] D. Shah, S. Patel, and S. K. Bharti, "Heart disease prediction using machine learning techniques," *SN Computer Science*, vol. 1, no. 6, pp. 1–6, 2020.

[27] M. Kumari and S. Godara, "Comparative study of data mining classification methods in cardiovascular disease prediction 1," 2011.

[28] G. Parthiban, A. Rajesh, and S. Srivatsa, "Diagnosis of heart disease for diabetic patients using naive bayes method," *Int. Journal Comput. Applications*, vol. 24, no. 3, pp. 7–11, 2011.

[29] A. F. Otoom, E. E. Abdallah, Y. Kilani, A. Kefaye, and M. Ashour, "Effective diagnosis and monitoring of heart disease," *Int. Journal Soft. Engg. Applications*, vol. 9, no. 1, pp. 143–156, 2015.

[30] D. Khanna, R. Sahu, V. Baths, and B. Deshpande, "Comparative study of classification techniques (SVM, logistic regression and neural networks) to predict the prevalence of heart disease," *Int. Journal Machine Learning Comput.*, vol. 5, no. 5, p. 414, 2015.

[31] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, "A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease," in *IEEE Symposium Comput. Commun. (ISCC)*, 2017, pp. 204–207.

[32] A. K. Dwivedi, "Performance evaluation of different machine learning techniques for prediction of heart disease," *Neural Computing and Applications*, vol. 29, no. 10, pp. 685–693, 2018.

[33] M. A. Khan, "An iot framework for heart disease prediction based on mdcnn classifier," *IEEE Access*, vol. 8, pp. 34 717–34 727, 2020.

[34] I. Tougui, A. Jilbab, and J. El Mhamdi, "Heart disease classification using data mining tools and machine learning techniques," *Health and Technology*, vol. 10, pp. 1137–1144, 2020.

[35] L. Sapra, J. K. Sandhu, and N. Goyal, "Intelligent method for detection of coronary artery disease with ensemble approach," in *Advances Commun. Comput. Technology*. Springer, 2021, pp. 1033–1042.

[36] R. Katarya and S. K. Meena, "Machine learning techniques for heart disease prediction: a comparative study and analysis," *Health and Technology*, vol. 11, no. 1, pp. 87–97, 2021.

[37] R. Bharti *et al.*, "Prediction of heart disease using a combination of machine learning and deep learning," *Computational Intelligence and Neuroscience*, vol. 2021, 2021.

[38] K. Divya, A. Sirohi, S. Pande, and R. Malik, "An iomt assisted heart disease diagnostic system using machine learning techniques," in *Cogn. Internet Med. Things Smart Healthcare*. Springer, 2021, pp. 145–161.

[39] M. R. Rana and N. Al-Musabbir, "Heart disease prediction using machine learning," 2021.