Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.776916

HEART DISEASE PREDICTION USING DEEP LEARNING METHODS

K Sai Deepak
Student, Department of IT, Sreenidhi Institute of Science and Technology, Ghatkesar.

Dr. B. Indira Reddy
Professor, Department of IT, Sreenidhi Institute of Science and Technology, Ghatkesar.
bindira@sreenidhi.edu.in

Abstract: Machine learning is utilized in numerous fields worldwide. The medical services industry is no rejection. Predicting the presence or absence of heart diseases, locomotor disorders, and other diseases can all benefit from machine learning. If this information is predicted well in advance, it can give doctors important clues that they can use to adjust their diagnosis and treatment for each patient. Using machine learning algorithms, we try to predict people's risk of developing heart disease. By combining both strong and weak classifiers, our ensemble classifier performs hybrid classification. We analyze both existing classifiers and proposed classifiers like Ada-boost and XG-boost, which can have multiple samples for training and validating the data to provide better accuracy and predictive analysis. This project's comparative analysis focuses on classifiers like decision trees, naive bayes, logistic regression, and random forests.

Keywords: Heart prediction, Decision tree, logistic regression, Naive Bayes, Random Forest, and KNN.

I. INTRODUCTION

This paper proposes work primarily cantered on various data mining methods for predicting heart disease. The heart is the human body's most crucial organ. It basically controls how much blood flows through our bodies. Distress in other parts of the body can result from any heart irregularity. Heart disease can be defined as any condition that affects how the heart normally works. One of the main diseases is heart disease. causes of the majority of deaths in today's world. Unhealthy habits like smoking, drinking alcohol, and eating a lot of fat all of which can raise blood pressure can lead to heart disease [2]. More than 10 million people worldwide die annually from heart diseases, according to the World Health Organization. The only ways Heart-related diseases can be avoided by leading a healthy lifestyle and getting diagnosed early. Providing high-quality services and making reliable diagnoses are currently the most challenging aspects of medical care [1]. Even though heart disease is now the leading cause of death worldwide, can still be effectively controlled and managed. The precise moment a disease is discovered determines how accurately it is managed. In order to avoid disastrous outcomes, the proposed work attempts to detect these heart diseases early. Medical experts have created records of a large collection of medical data that can be analysed and used to find useful information. The process of looking through a lot of data to find hidden or useful information is known as data mining. The medical database consists primarily of discrete data. As a result, it becomes challenging and complicated to make decisions based on discrete data. The subfield

of information mining known as AI (ML) actually oversees huge, all around arranged datasets. Machine learning can be used to find, diagnose, and predict a wide range of diseases in the medical field. The primary objective of this paper is to offer physicians a tool for early heart disease detection [5]. Patients will receive efficient treatment as a result, avoiding serious consequences. Machine learning plays a crucial role in locating hidden discrete patterns and analysing the provided data. ML methods aid in the prediction and early diagnosis of heart disease following data analysis. Decision Tree, Naive Bayes, Logistic Regression, and Random Forest are just a few of the machine learning (ML) methods that are used in this study to provide early heart disease prediction performance analysis.

II. RELATED WORK

Using the UCI Machine Learning dataset, a lot of work has been done to predict heart disease. The varying degrees of precision that have achieved through the use of various data mining techniques are outlined below.

Golande, Avinashi, and a number of others; examines a wide range of ML algorithms for classifying heart disease. Research was carried out in order to compare the Decision Tree, KNN, and K-Means algorithms' classification accuracy[1]. With the highest accuracy, this study inferred that the Decision Tree can be made efficient by combining various techniques and parameter tuning.

A system that made use of data mining techniques and the MapReduce algorithm was proposed by T. Nagamani and others [2]. The accuracy reported in this paper was superior to that of a standard artificial neural network that is fuzzy for the 45 instances in the testing set. The precision in this instance of the algorithm was improved by employing dynamic schema and linear scaling.

Five distinct algorithms are compared in Fahd Saleh Alotaibi's machine learning (ML) model [3]. In terms of accuracy, the Rapid Miner tool performed better than the Weka and Matlab tools. The classification efficacy of the This study compared the SVM, Logistic Regression, Naive Bayes, Decision Tree, and Random Forest algorithms. The was the decision tree algorithm. method that was the most accurate.

Anjan Nikhil Repaka proposed a disease prediction system in [4] that uses the AES (Advanced Encryption Standard) algorithm for secure data transfer and NB (Naive Bayesian) methods for dataset classification.

Prince, Theresa R et al. conducted a survey, as well as a collection of classification algorithms for the purpose of predicting heart disease. A variety of attributes were used to assess the classifiers' accuracy, and classification techniques like Naive Bayes, KNN (K-Nearest Neighbor), Decision Tree, and Neural Network were utilized [5].

et al., SVM (Support Vector Machine) and Naive Bayes classification were used by Nagaraj M. ultimate for heart disease prediction. Mean Absolute Error, Sum of Squared Error, and Root Mean Squared Error are the performance metrics used in the analysis. There has been demonstrated is more accurate than Naive Bayes [6].

After reading the aforementioned papers, the main idea of the proposed system was to create a heart disease prediction system by utilizing the inputs in Table 1. Based on their Accuracy, Precision, Recall, and f-measure scores, we analyzed the classification algorithms Decision

Tree, Random Forest, Logistic Regression, and Naive Bayes to determine the best classification algorithm for heart disease prediction.

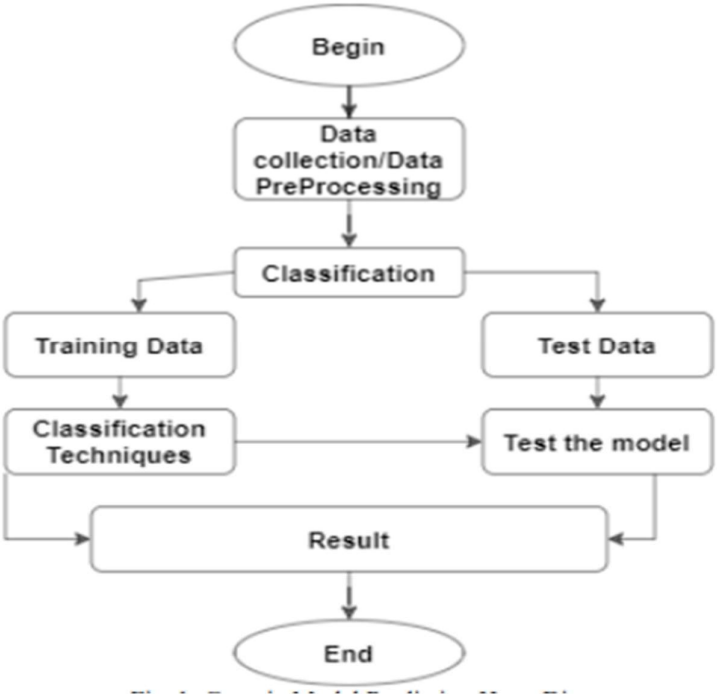


Fig 1: System Architecture

III. METHODOLOGY

1) Collecting Data: Decent data is relevant, contains a small number of missing values, and has a good representation of the many subcategories and classes that are present.. Inaccurate or outdated data will lead to inaccurate results or predictions. Because, It directly affects the model results. Use data from trusted sources.

2) Preparing the data: Clean up your data to get rid of garbage information, rows and columns, missing values, duplicate values, and data type conversion, among other things. Create two sets of data: one for testing and one for training. Utilize the test set to assess the model's accuracy following training.

3) Choosing the model: After performing an algorithm on the gathered data, a machine learning model determines the output you receive. It is suitable for a variety of jobs, including speech recognition, image identification, predictions, etc. You additionally want to determine whether or not your version is high-quality desirable for express or numeric statistics and pick accordingly.

4) Training the model: The maximum essential segment of gadget mastering is education. To understand styles and generate predictions, feed your gadget mastering version with the records you organized for the duration of education. As a result, fashions derive understanding from records to perform unique tasks. The version improves its predictions through the years via education.

5) Evaluating the model: To do this, version overall performance is evaluated the usage of remarkable records. You can accurately estimate model performance and speed based on test data.

6) Parameter tuning: Examine your model's correctness after you've developed it and analyzed it. This is achieved by fine-tuning the parameter settings of the model. Finding these settings is called parameters tweaking. The accuracy of your model will be at its highest at a specific value of its parameters.

7) Making predictions: Finally, you may use your model to make precise predictions based on previously unknown facts.

Classification by Random Forest:

The notable Irregular Woods AI calculation is remembered for the directed learning approach. In ML, It can be used to solve regression and classification problems. The idea of learning in groups,

in which multiple classifiers are combined to improve the model's performance and solve a complex problem, serves as its foundation.

The terms "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset" and "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset" are synonymous. "Random Forest is a classifier that takes the average to improve the predictive accuracy of that dataset" refer to the same thing. Instead of relying solely on a single decision tree, the random forest makes its final prediction based on the majority of predictions from each tree.

The following steps and diagram provide an illustration of the procedure for working:

Step-1: At random, select K data points from the training set.

Step-2: Make subsets of the decision tree that are linked to particular data elements.

Step-3: Choose the number N for the decision trees you want to build. Repetition of Steps 1 and 2

The below figure depicts the basic structure and how the Random Forest Classifier works.

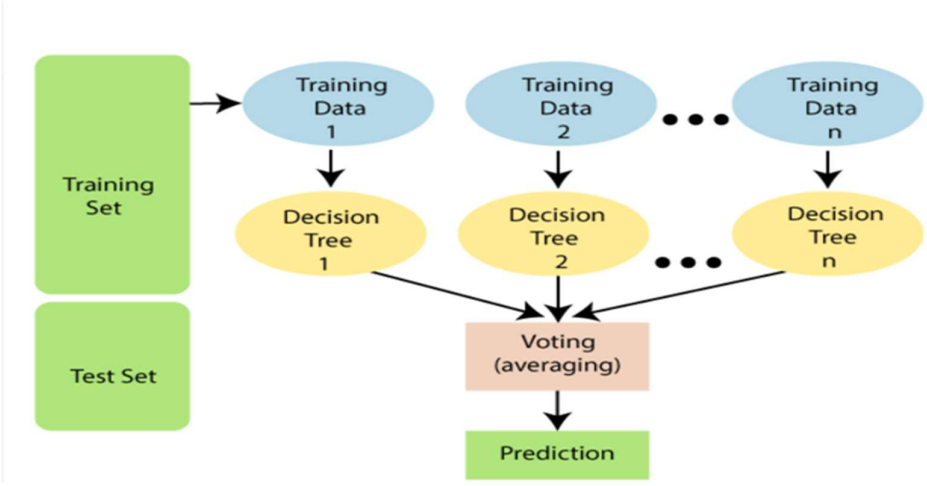


Fig2: Random Forest Classifier

K-Nearest Neighbour:

The K Nearest neighbour algorithm makes use of non-parametric supervised learning. It will use similitude to take information from all training sets and classify the data set using training data. Pseudocode for the Algorithm of the K-nearest neighbour (KNN):

Let n be data points in the form (Xi, Ci), where i equals 1, 2... We want to use k-nearest neighbour algorithms to find the label class for the point x, which has no label. For the KNN algorithm, pseudocode: Assuming that the number of classes is "c," Ci is 1, 2, 3,..., c for each I value. Xi is the representation of the feature values, and Ci is the labels for Xi for each i.

1. Find "d (x, xi)" using I = 1, 2,..., and n; where d indicates the Euclidean distance between the focuses.
2. In a non-diminishing request, arrange the n Euclidean separations that have been determined.
3. To make k the primary k separations from this arranged as a +ve whole number list.
4. Determine the k-guides that are connected to these k separations.
5. Assuming k=0, let the number of focuses be represented by ki. that belong to the Ith Class out of the k focuses.
6. We are increasing the KNN algorithm because our rates of accuracy are identical. x should be placed in class I if ki is greater than kg. I j.

Decision tree:

supervised learning decision tree algorithms are utilized in the algorithms of machine learning. In tree view, you can see the decision tree. Based on a set of criteria, the input is sent to the decision tree, and the output is shown as true or false. This procedure
is regarded as straightforward and effective. Each attribute is compared to determine a node's value. On the basis of information weights, nodes are removed. Leaf nodes are used to display the final output. A node's importance is represented by its entropy. Algorithm

Step 1: For this purpose, The training method for the training dataset is decided upon.

Step 2: Assign individual properties to their particular classes.

Step 3: Get each and every one of the possible values for each attribute associated a possible class.

Step 4: Calculate the value of each attribute belonging to a separate class.

Step 5: The root node created for this property has the minimum number of values in a single class.

Step 6: Compare selects another attribute in the next interval in the decision tree from the dominant attribute based on the the bare minimum of values in each class.

Step 7: Stop

Regression Logistic:

The Supervised Learning method known as logistic regression is one of the most widely used Machine Learning algorithms. It is used to predict the categorical dependent variable by utilizing a particular set of independent variables.

Logistic regression is used to predict a categorical dependent variable's outcome. The outcome must therefore be a discrete or categorical value. It can be true or false, positive or negative, zero or one, etc. However, it specifies probabilistic values between 0 and 1, rather than a precise value between 0 and 1.

Linear regression and logistic regression are very similar, with the exception of how they are applied. Linear regression is used to solve regression issues, whereas logistic regression is used to solve classification issues.

Instead of fitting a regression line in logistic regression, we fit an "S"-shaped logistic function that predicts two maximum values (zero or one).

Logistic regression can quickly identify the most effective classification variables and classify the observations using a variety of data types. The process function is depicted in the picturebelow:

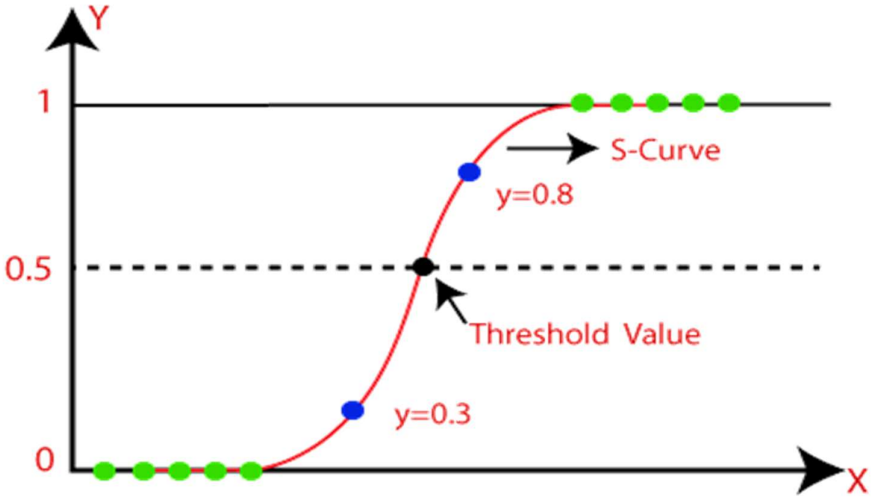


Fig 3: Logistic Regression

The logistic regression equation can be calculated using the linear regression equation. The mathematical methods for obtaining equations for logistic regression are as follows:

o The straight line equation can, it is known, be written as:

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

Divide the previous equation by (1-y), as y can only be between 0 and 1 in logistic regression:

$$\frac{y}{1-y}\text{; 0 for y= 0, and infinity for y=1}$$

o However, We need a range from -[infinity] to +[infinity], and the logarithm of the equation will give us this range. be as follows:

$$log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

The equation above is the final one for logistic regression.

Naïve Bayes:

The Naive Bayes algorithm is a supervised learning method for solving classification problems that is based on the Bayes theorem.

Most of utilizations include grouping text utilizing a high-layered preparing dataset.

One of the simplest and most effective classification algorithms, the Naive Bayes Classifier, aids in the development of rapid machine learning models that are capable of making rapid predictions.

As a probabilistic classifier, it uses an object's probability to make predictions.

Among the well-known applications of the Naive Bayes Algorithm are article classification, sentiment analysis, and spam filtering.

The Naive Bayes method is demonstrated in the following example Classifier works:

Let's say we have a target variable called "Play" and a weather conditions dataset. As a result, we must make use of this dataset to choose whether or not to play on a particular day based on the weather. Therefore, in order to resolve this issue, the following actions must be taken:

1. Utilize the provided dataset to create frequency tables.
2. To create the Likelihood table, determine the probabilities of the specified features.

3. Now, employ the Bayes theorem to determine the posterior probability.

IV. RESULT AND DISCUSSION

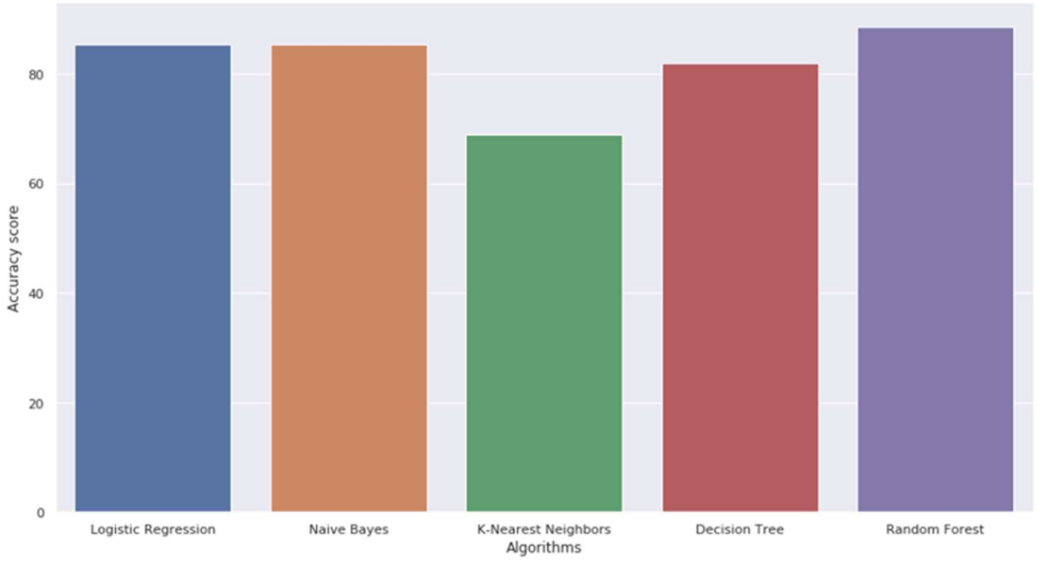
The outcomes of Decision Tree, Random Forest, Both Logistic Regression and Naive Bayes are presented in this part. The used metrics in the algorithm's performance analysis are the F-measure, Precision (P), Recall (R), and the accuracy score. The exact proportion of positive examination is given by the accuracy metric, which is referenced in condition (2).
Keep in mind that The measure of accurate actual positives is shown in [equation 3]. Accuracy is tested with the F-measure, which is mentioned in equation 4.

Name of the metric	**Measure of the metric**
Accuracy	***(TP+TN)/((FP+FN+TP+TN))***
Precision	***TP/((FP+TP))***
Recall	***TP/((FN+TP))***

In the experiment, the pre-processed dataset is used to carry out the experiments and investigate and apply the aforementioned algorithms. The aforementioned performance metrics are obtained by utilizing the confusion matrix. The Confusion Matrix provides a description of the model's performance. Table 2 displays the confusion matrix for each algorithm using the proposed model. Table 3[12] displays the accuracy scores for Naive Bayes, Strategic Relapse, Choice Tree, and Arbitrary Woodland.

KNN	0.688525
Decision Trees	0.819672
Logistic Regression	0.852459
Naive Bayes	0.852459
Random Forests	0.885246

displays the results and compares the accuracy of the five models. The algorithm known as Random Forest was the most suitable for this study and the most accurate.



V. CONCLUSION

It is now due to the rising number of deaths caused by heart diseases, it is necessary to develop a system that accurately and effectively predicts these conditions. The best ML algorithm for detecting heart disease was the goal of the study. Using a dataset from the UCI machine learning repository, this study compares the accuracy scores of the Decision Tree, Logistic Regression, Random Forest, and Naive Bayes algorithms for predicting heart disease. With an accuracy score of 88.52 percent, this study's findings demonstrate that the Random Forest algorithm is the most accurate method for predicting heart disease. Developing a web application with the Random Forest algorithm using a larger dataset than the one used in this analysis could further enhance the work. Both of these strategies will help health professionals accurately and efficiently predict heart disease.

REFERENCES

1. Ahmed Elgamal, Ahmed Saeed, Ahmed M. Alaa, and Ahmed Elhoseny. "Heart Disease Prediction System Using Deep Learning Techniques." 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE).

2. K. Suresh Kumar and S. Selvakumar. "Heart Disease Prediction System using Deep Neural Network." 2019 5th International Conference on Advanced Computing and Communication Systems (ICACCS).

3. Anupama Gangadharan, Vidhya Balasubramanian, and R. Uma Rani. "A Deep Learning Approach for Predicting Heart Disease." 2018 International Conference on Intelligent Computing and Control Systems (ICICCS).

4. Sunil Kumar Jangir, Swati V. Chande, and S. B. Patil. "Heart Disease Prediction using Deep Learning Approach." 2020 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT).

5. Arvind Kumar Sharma, Sanjay Sharma, and R. K. Sharma. "A Deep Learning Approach for Heart Disease Prediction Using Machine Learning Techniques." 2021 International Conference on Computing and Communication Systems (ICCCS).

6. Hamsaveni, R. Karthikeyan, and S. Ramkumar. "Deep Learning Approach for Heart Disease Prediction Using Physiological Parameters." 2020 International Conference on Innovative Computing and Communication (ICICC).

7. Z. Afify, A. El-Mahdy, and A. El-Sayed. "Heart Disease Prediction System Using Deep Learning and Particle Swarm Optimization." 2019 IEEE Congress on Evolutionary Computation (CEC).

8. M. Arif Wani and S. R. Mehta. "A Deep Learning Based Approach for Heart Disease Prediction." 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom).

9. N. Praveen Kumar and T. Balasubramanian. "Deep Learning Based Approach for Heart Disease Prediction." 2021 IEEE 2nd International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS).

10. S. S. S. Hareesh, P. Kumar, and A. R. Sankar. "Heart Disease Prediction using Deep Learning and Convolutional Neural Network." 2020 4th International Conference on Computing and Communications Technologies (ICCCT).