ANN-Driven Feature Selection for Improved Heart Disease Classification via Machine Learning and Web-based Application

by

MD. SHAMEEM AHAMMED
19102913

MD. BORHAN UDDIN
19102936

A thesis submitted to the
Department of Electrical and Electronic Engineering
in partial fulfillment of the requirements for the degree of
BACHELOR OF SCIENCE IN
ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
Jatiya Kabi Kazi Nazrul Islam University
Mymensingh, Bangladesh

December 2024

Acknowledgement

First and foremost, we thank the Almighty for allowing us to finish our thesis without any serious setbacks.

Second, we want to thank our supervisor, Dr. Md. Mahbubur Rahman, Associate Professor of Electrical and Electronic Engineering at Jatiya Kabi Kazi Nazrul Islam University, for his continual support, supervision, and advice during this study. Completing this work would have been difficult without his invaluable guidance, suggestions, and gracious support. His passion for teaching and research provided us with exciting opportunities to broaden our scientific knowledge and pique our interest in the field of Artificial Intelligence.

All of those who directly and indirectly assisted us in carrying our work is deeply appreciated. Finally, we wish to express our gratitude to our family members for their emotional support. We would not have gotten where we are now without them.

Abstract

Heart disease is a leading cause of global mortality, improving patient outcomes requires early identification and precise categorization. Due to human error, the majority of traditional diagnostic techniques remain time-consuming, expensive, and inaccurate. The following study suggests utilizing an **ANN-driven feature selection-based** approach in combined with machine learning approaches to deal with heart diseases, identify them, and further categorize them by degree, including mild, moderate, severe, and critical..

In order to eliminate superfluous features, encode categorical variables, and impute missing values, the dataset, comprising 15 features and 920 instances. To address the shortage of data, augmentation techniques were employed to expand the dataset to 8,000 events Exploratory data analysis, or EDA, provided the understanding of feature patterns and relationships with the goal variable. Feature engineering using an artificial neural network (ANN) reduced dimensionality to 8 key features.

For classification, Artificial Neural Network (ANN) and six machine learning models—Random Forest, XGBoost, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Logistic Regression, and LightGBM—were trained and evaluated using an 80:20 train-test split. The Random Forest model was chosen for deployment because it had the best accuracy, precision, recall, and F1 score. Bar charts for model comparison and confusion matrices were used to display the classification results.

The model was deployed as a web-based application using Streamlit, enabling real-time prediction of heart disease presence and classification of its severity. This dual-phase prediction system provides a scalable and effective way to close the gap between conventional diagnostic techniques and AI-driven healthcare solutions.

Future work involves integrating real-time clinical data and expanding the system to include multi-disease predictions. This research demonstrates the transformative potential of AI in healthcare, providing a robust and accessible framework for heart disease prediction and classification.

Keywords: Heart Disease Classification, Deep Learning, Machine Learning, ANN, Random Forest, XGBoost, Support Vector Machine, K-Nearest Neighbor, Logistic Regression, LightGBM.

Declaration

It is hereby declared that

1. The thesis we submitter was written as part of our degree program at Jatiya Kabi Kazi Nazrul Islam University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material that has been accepted, or submitted, for any other degree or diploma at a university or other university.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

MD. Shameem Ahammed 19102913	Md. Borhan Uddin 19102936

Approval

The thesis titled “ANN-Driven Feature Selection for Improved Heart Disease Classification via Machine Learning and Web-based Application” submitted by

1. MD. Shameem Ahammed (19102913)
2. Md. Borhan Uddin (19102936)

On January 9, 2025 has been accepted as satisfactory in the partial fulfillment of the requirement for the Bachelor of Science in Electrical and Electronic Engineering on that date.

Supervisor:

_____

Dr. Md. Mahbubur Rahman
Associate Professor
Department of
Electrical and Electronic Engineering
Jatiya Kabi Kazi Nazrul Islam University

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AI Artificial Intelligence

ML Machine Learning

DL Deep Learning

ANN Artificial Neural Network

RF Random Forest

XGB XGBoost

SVM Support Vector Machine

KNN K-Nearest Neighbor

LR Logistic Regression

LGB LightGBM

TN True Negative

TP True Positive

FN False Negative

FP False Positive

Chapter 1

Introduction

1.1 Artificial Intelligence into Medical Science

Artificial intelligence (AI) is a field of science and engineering that focuses on creating computer programs that can exhibit intelligent behavior. It is a fundamental philosophical tenet that artificial intelligence (AI) is a greater intelligence enhancement. People's intelligence with machines is the process of augmenting people's intellect with machines that have enough intelligent problem-solving capabilities that the combination of the human and machine is superior to either alone. Modern medicine faces the difficulty of gathering, analyzing, and using the vast quantities of knowledge required to treat complicated clinical problems. The development of medical artificial intelligence has been tied to the creation of AI programs designed to aid clinicians in formulating diagnoses, making therapeutic decisions, and predicting patient outcomes. To aid clinicians in making diagnoses, treatment decisions, and predicting patient outcomes, AI programs have been developed. These programs, including machine learning, deep learning in data science, aim to simplify data and knowledge management and aid healthcare professionals in their daily work[1].

According to the World Health Organization (WHO), heart disease is a leading cause of death worldwide, accounting for over 17.9 million deaths each year[2]. Improving patient outcomes and lowering healthcare costs depend on early identification and precise classification of cardiac disease. However, traditional diagnostic techniques—which depend on imaging, laboratory testing, and clinical examinations—are time-consuming, costly, and prone to human error. These challenges highlight the important role of scalable and automated solutions, particularly in areas with restricted access to cutting-edge medical facilities.

Artificial intelligence (AI) and machine learning (ML) have emerged as transformative technologies in healthcare, offering the potential to revolutionize disease prediction and classification. Compared to traditional approaches, AI-driven systems are capable to evaluate massive data, spot subtle trends, and produce predictions with greater efficiency and precision. ML algorithms like Random Forest, Support Vector Machines (SVM), and XGBoost have demonstrated potential in identifying cardiac disease. Furthermore, deep learning methods like as Artificial Neural Networks (ANNs) are excellent at extracting features, which makes it possible to create reliable models for challenging prediction challenges..

1.2 Motivation

The motivation for this study stems from the growing global burden of heart disease and the limitations of traditional diagnostic approaches. Patients' risks increase and solutions take longer as a result of current approaches' frequent inability to offer accurate and timely assessments of illness severity. A viable alternative that offers precise and scalable techniques for the early diagnosis and categorization of heart disease is the combination of AI and ML. Additionally, the dearth of simple to use instantaneously prediction tools drives the design of a web-based application that is usable by both patients and medical professionals.

1.3 Aim

The goal of this research endeavor is to develop an extensive AI-driven system that can determine if a patient has cardiac disease and classify it according to severity into four levels: critical, severe, moderate, and mild.

1.4 Objectives

1. **Data Generation:** To improve a small dataset so that it may be used for model evaluation and training.
2. **Adaptive Supportive tool:** Providing medical personnel with a responsive helpful tool that will enable them to identify and categorize a patient's cardiac status..

1.5 Contribution

This study introduces a novel framework integrating ANN-driven feature selection with ML-based classification for predicting and classifying heart disease severity. In contrast to traditional methods that only use binary prediction, this study uses a multi-stage categorization system that offers information on how the illness develops. The deployment of the model as a web-based application ensures accessibility and scalability, addressing the practical challenges of implementing AI in real-world healthcare settings.

By bridging the gap between advanced AI techniques and practical deployment, this research contributes to the growing body of work on AI in healthcare. It highlights the potential of integrating ML and DL methodologies to create robust, accessible, and impactful diagnostic tools for heart disease. Future extensions include incorporating real-time clinical data and expanding the application to cover other critical diseases, further enhancing its utility in modern medicine.

1.6 Thesis Structure

Chapter 1 introduced key aspects, including the integration of artificial intelligence into medical science, the study's motivation, its objectives and key contribution.

Chapter 2 delved into the foundational concepts of machine learning, covering its basics, classification, deep learning, applications in various fields, and its relevance to heart disease.

Chapter 3 conducted a comprehensive literature review, providing an overview of existing knowledge and research on heart disease prediction and classification.

Chapter 4 outlined the proposed work, explained the methodology employed, describing the step-by-step process used to conduct the research, including data analysis and model development.

Chapter 5 presented the results and outcomes derived from the research, showcasing the findings and achievements of the study.

Chapter 6 wrapped up the thesis, offering a conclusion summarizing the key insights and accomplishments. Additionally, it outlined areas for future work, providing a roadmap for further research in the field.

Chapter 2

Background Analysis

2.1 Fundamental of Machine Learning

Machine learning is a subject of computer science that focuses on the development of algorithms that rely on a collection of examples of some event in order to be useful. These examples may originate from nature, be human-made, or be generated by another program[3].

The process of solving a practical problem by 1) collecting a dataset and 2) algorithmically constructing a statistical model based on that dataset is another definition of machine learning.

2.2 Types of Learning

Learning are four types. These are supervised, semi-supervised, unsupervised and reinforcement learning.

2.2.1 Supervised Learning

In supervised learning, the dataset is the collection of labeled examples $(xi, yi)_{i=1}^{N}$. Each element x_i among N is called a feature vector. A feature vector is a vector in which each dimension j = 1, . . . ,D contains a value that describes the example somehow. That value is called a feature and is denoted as $x^{(j)}$. For instance, if each example x in our collection represents a person, then the first feature, $x^{(1)}$, could contain height in cm, the second feature, $x^{(2)}$, could contain weight in kg, $x^{(3)}$ could contain gender, and so on. For all examples in the dataset, the feature at position j in the feature vector always contains the same kind of information. It means that if $x_i^{(2)}$ contains weight in kg in some example x_i, then $x^{(2)}$ k will also contain weight in kg in every example x_k, k = 1, . . . ,N. The label y_i can be either an element belonging to a finite set of classes 1, 2, . . . ,C, or a real number, or a more complex 6 structure, like a vector, a matrix, a tree, or a graph. For instance, if your examples are email messages and your problem is spam detection, then you have two classes spam, not-spam[3].

The goal of supervised learning is to use the dataset to build a model that takes a feature vector x as input and provides information that allows the label to be inferred for this feature vector. The likelihood that a person has cancer, for example, may be generated by the model built from the dataset for people using a feature vector characterizing the individual as input. Regardless of depth, supervised learning is the most common kind of machine learning. Let's say we are attempting to build a system that can recognize images of a human, a pet, a car, or a home. We start by collecting a vast collection of images, each of which is tagged with the category to which it belongs. The system creates a vector of scores after training, with a distinct score for each category. It is uncommon for the targeted category to have the highest score before training. An arbitrary function is used to calculate the variance (or distance) among the obtained scores and the intended scoring pattern. To lessen this error, the system then modifies its internal settings. These weights, which are actual numbers, might be thought of as "knobs" that specify the machine's input-output

function. There may be hundreds of millions of these adjustable weights and hundreds of billions of instances that might be labeled in a typical deep learning system.

2.2.2 Unsupervised Learning

In unsupervised learning, the dataset is a collection of unlabeled examples $\{xi\}_{i=1}^{N}$. Again, x represents a feature vector. The objective of an unsupervised learning method is to develop a model that accepts a feature vector x as input and converts it into a value that may be utilized to address a real-world issue or into another vector. When clustering, for instance, the model gives back the cluster's id for every feature vector in the dataset. A feature vector with fewer characteristics than the input x is the model's output in reducing dimensionality, whereas a real number describing how x differs from a "typical" sample in the collection of data is the outcome in outlier identification [3].

2.2.3 Semi-Supervised Learning

In semi-supervised learning, the training dataset includes both labeled and unlabeled samples. In the majority of instances, there are far more unlabeled samples than recognized ones. The goal of methods for semi-supervised learning is the same as that of supervised learning. It is hoped that using an excessive amount of unlabeled instances would help the learning process find (or calculate) a more accurate model[3].

2.2.4 Reinforcement Learning

Reinforcement learning is a subfield of machine learning in which the machine “lives” in an environment and is capable of sensing the environment’s state as a vector of features. The machine can do tasks in any condition. In addition to producing different rewards, different actions can also change the state of the machine's surroundings. A policy is intended to be learned using a reinforcement learning algorithm. A policy is a function f that takes as input the feature vector of a state and returns the best course of action to take regarding that state (much like the model in supervised learning). An action qualifies as perfect if it achieves the expected average payout [3].

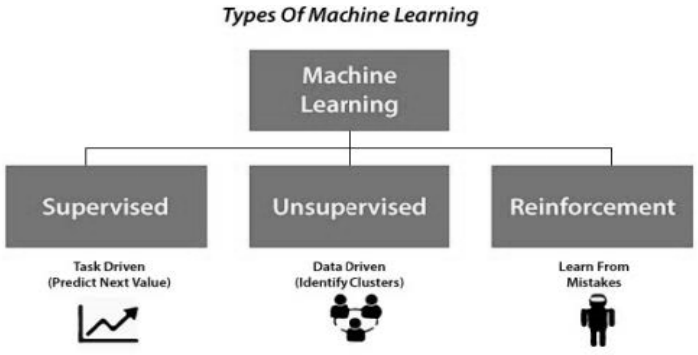

Figure 1: Types of Machine Learning

2.3 Classification vs. Regression

Classification is the challenge of assigning a label to an unlabeled instance automatically. Spam detection is a well-known illustration of classification. An classification learning algorithm resolves the classification problem in machine learning via a collection of labeled examples as inputs. This algorithm then develops a model that can utilize an unlabeled example as input and either directly output a label or output a number that the data analyst can use to quickly determine the label. One example of a number like this is probability. Classifying problems include grouping labels into several groups, as the name implies. Binary classification, also known as binary classification, occurs when there are just two categories to select from, such as "sick" against "healthy," "spam" versus "not spam," etc. Multiple classes are utilized in multiclass classification, frequently referred to as multinomial classification. By itself, many learning algorithms allow more than two classes, whereas others are binary classification algorithms by design. It is possible to convert a binary classification learning method into a multiclass one using certain strategies[3].

Regression is a problem where a real-valued label (also known as a target) must be forecasted from an unlabeled sample. The calculation of property price evaluation based on house features, such as space, number of rooms, location, and more, is a well-known example of regression. The regression problem can be solved employing a regression learning approach. This method creates a model that can take an unlabeled example as input and output a target by employing a set of labeled circumstances as inputs. The regression problem may then be resolved using the unlabeled example**.** [3].

2.4 Fundamental Algorithms

In this section, I explain five algorithms that are not just the most well-known but also extremely effective on their own or as building blocks for the most effective learning algorithms now available.

2.4.1 Linear Regression

Linear regression is a well-known regression learning approach that builds a model that is a linear combination of the input example's features

We have a collection of labeled examples $(xi, yi)_{i=1}^{N}$ where N is the size of the collection, x_i is the D-dimensional feature vector of example i = 1, . . . ,N, y_i is a real-valued target and every feature $x_i^{(j)}$, j = 1, . . . ,D, is also a real number. We want to build a model $f_{w,b}(x)$ as a linear combination of features of example x:

$$fw, b(x) = wx + b \qquad Eq(1)$$

where b is a real value and w is a D-dimensional vector of parameters. The model f is parametrized by two values, w and b, as shown by the notation $f_{w,b}$. For a given x, we will utilize the model to predict the unknown y as follows: $y \leftarrow f_{w,b}$. When used for the same example, two models parametrized by two independent pairings (w, b) will probably provide two different predictions. Finding the ideal values (w∗, b∗) is our goal. It goes

without saying that the model that produces the most accurate forecasts is defined by the ideal parameter values.[3].

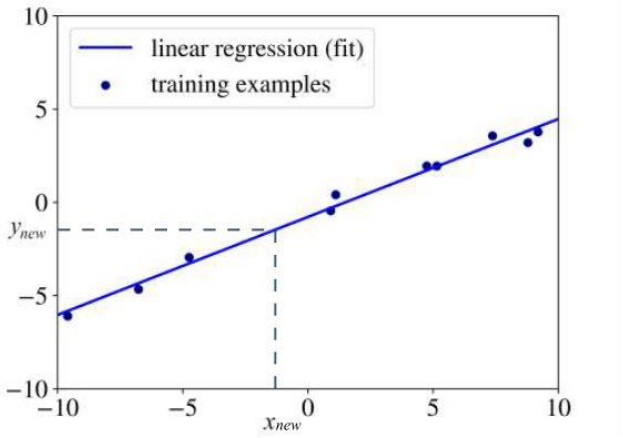

Figure 2: Linear Regression for one-dimensional examples

Figure 2 shows the regression line for one-dimensional samples (dark-blue dots) in light-blue. For a new unlabeled input sample x_{new}, we may utilize this line to forecast the value of the target y_{new}. The main distinction with the one-dimensional case is that, if our examples are D-dimensional feature vectors (for D > 1), the regression model is a plane (for two dimensions) or a hyperplane (for D > 2) rather than a line.

To get this latter requirement satisfied, the optimization procedure which we use to find the optimal values for $w*$ and $b*$ tries to minimize the following expression [3]:

$$\frac{1}{N}\sum_{i=1.....N}(f_{w,b}(x_i) - y_i)^2 \qquad Eq(2)$$

In mathematics, the expression we minimize or maximize is called an objective function, or, simply, an objective. The expression $(f(x_i)-y_i)^2$ in the above objective is called the **loss function**. It is a form of punishment for incorrectly classifying example I. Squared error loss is the name given to this specific loss function selection. Every model-based learning technique has a loss function, and in order to identify the optimal model, we attempt to minimize the cost function. The average loss, also known as empirical risk, provides the cost function in linear regression. The average of all penalties incurred when a model is applied to training data is known as the model's average loss, or empirical risk.

2.4.2 Logistic Regression

First, it is important to note that logistic regression is not a regression, but rather a categorization learning technique. The name is derived from statistics, as the mathematical formulation of logistic regression is comparable to that of linear regression. In logistic regression, we still want to model y_i as a linear function of x_i, however, with a binary y_i this

is not straight forward. The linear combination of features such as $wx_i + b$ is a function that spans from minus infinity to plus infinity, while y_i has only two possible values.

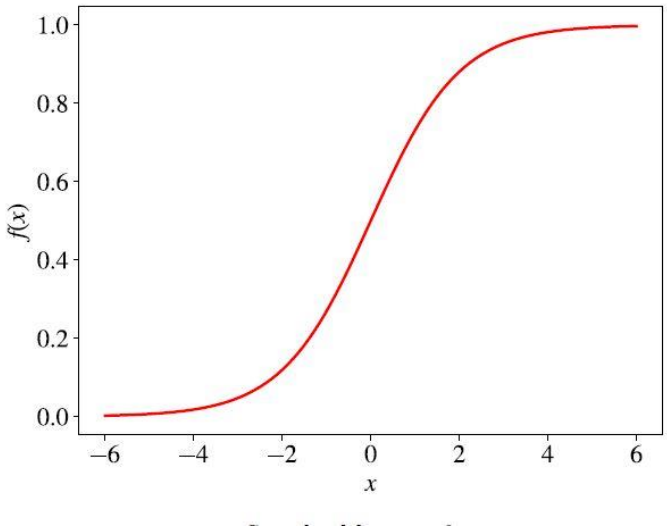

Figure 3: Standard logistic function

Scientists were keen to discover a linear classification model at the time, when the lack of computers forced them to do computations by hand. They discovered that all we would have to do is identify a simple continuous function with a co-domain of (0, 1) if we denoted a negative label as 0 and a positive label as 1. In this situation, we give x a negative label if the value the model returns for input x is closer to 0; if not, the sample is labeled as positive. The standard logistic function, also referred to as the sigmoid function, is one function that satisfies this attribute[3]:

$$f(x) = \frac{1}{1 + e^{-x}} \qquad Eq(3)$$

where e is the base of the natural logarithm (also called Euler's number; e^x is also known as the exp(x) function in Excel and many programming languages). Its graph is visualized in fig 3.

We can determine how well the typical logistic function matches our classification goal by examining the the figure: if we correctly adjust the values of x and b, we can interpret the result of f (x) as the likelihood that yi will be positive. For example, we would state that the class of x is positive if it is bigger than or equal to the threshold of 0.5; if not, it is negative. According to the issue, the threshold may be defined significantly in practice. So the logistic regression model looks like this:

$$f(x) = \frac{1}{1 + e^{-(wx+b)}} \qquad Eq(4)$$

The well-known formula wx + b is visible in linear regression. Now, how can we determine which values $w*$ and $b*$ are optimal for our model? The average squared error loss, also referred to as the mean squared error or MSE, was the empirical risk that we reduced in linear regression..

In logistic regression, we optimize a probability of our training set based on the model instead of use a squared loss and aiming to lower the empirical risk. The probability function in statistics shows how likely an observation (an example) is based on our model[3].

2.4.3 Decision Tree

An acyclic graph called a decision tree can assist you in making decisions. A particular feature j of the vector of parameters is examined at each branching site of the graph. If the value of the feature is below a certain threshold, the left branch is taken. If the value is above the threshold, the right branch is taken. When the leaf node is reached, the class to which the example belongs is decided.

A decision tree can be learned from data. As before, we have a set of instances with labels; the labels are in the range 0 to 1. In order to forecast the class of an example given a feature vector, we wish to construct a decision tree. The decision tree learning algorithm has several different formulations. We only look at one in this section, ID3 [3].

The optimization criterion, in this case, is the average log-likelihood:

$$\frac{1}{N}\sum_{i=1.....N} y_i ln f_{ID3}(x_i) + (1 - y_i ln\,(1 - f_{ID3}(x_i)) \qquad Eq(5)$$

Where f_{ID3} is a decision tree.

It now resembles logistic regression quite a bit. Nevertheless, the ID3 approach represents the optimization criterion by building a non-parametric model, opposing to the logistic regression learning algorithm, which creates a parametric model $f_{w*,b*}$ by finding an optimal solution to it, the ID3 algorithm optimizes it approximately by constructing a non-parametric model $f_{ID3}(x) = Pr(y = 1|x)$.

2.4.4 Support Vector Machine

Every feature vector is seen by SVM as a location in a high-dimensional space, which in this case has 20,000 dimensions. An imagined 20,000-dimensional line, or hyperplane, is generated by the algorithm to divide samples with positive labels from examples with negative labels after placing all feature vectors on an unreal 20,000-dimensional plot. The decision margin is the line that splits instances of several classes in machine learning [3].

Two parameters provide the hyperplane's equation: a real number b like this and a real-valued vector w with the same dimensions as our input feature vector x [3]:

$$wx - b = 0 \qquad Eq(6)$$

where the expression wx means $w^{(1)}x^{(1)} + w^{(2)}x^{(2)} + ... + w^{(D)}x^{(D)}$, and D is the number of dimensions of the feature vector x.

Now, the predicted label for some input attributes vector x is given like this:

$$y = sign(wxb) \qquad Eq(7)$$

where sign is a mathematical operator that takes any value as input and gives +1 if the input is a positive number or -1 if the input is a negative number.

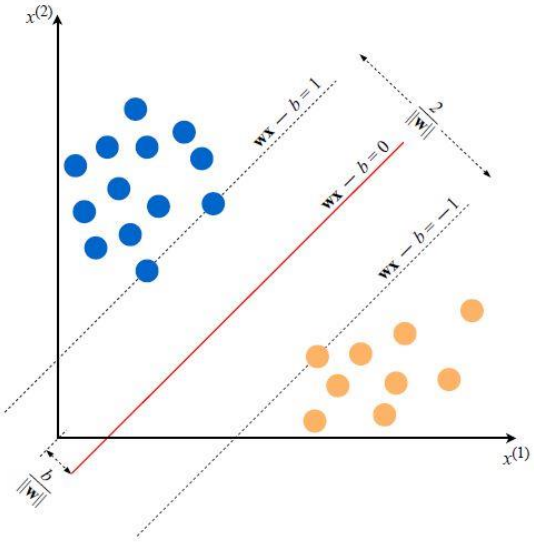

Figure 4: SVM model for two-dimensional feature vectors

The learning of Support Vector Machine algorithm is to leverage the dataset and find the optimal values w^* and b^* for parameters w and b. After identifying these optimal values, the model f(x) is then defined as[3]:

$$f(x) = sign(w * x - b *) \qquad Eq(8)$$

Consequently, in order to use an SVM model to predict whether an email message is spam or not, you must take the text of the message, turn it into a feature vector, multiply it by $w*$, remove $b*$, and then determine the sign of the result. The prediction will be as follows: +1 indicates "spam," and -1 indicates "not-spam.". Machine solves an optimization problem to find $w*$ and $b*$. The constraints are naturally:

- $wxi - b \geq 1 \, if \, yi = +1$, and
- $wxi - b \leq 1 \, if \, yi = -1$

Additionally, the hyperplane to have the widest gap between positive and negative instances. According to the decision boundary, the margin is the separation between the nearest instances of two classes. [3].

Two critical questions need to be answered:

1. What happens if the data comprises noise and no hyperplane is able to accurately distinguish between positive and negative examples?
2. What if a higher-order polynomial could be used to separate the data instead of a plane?

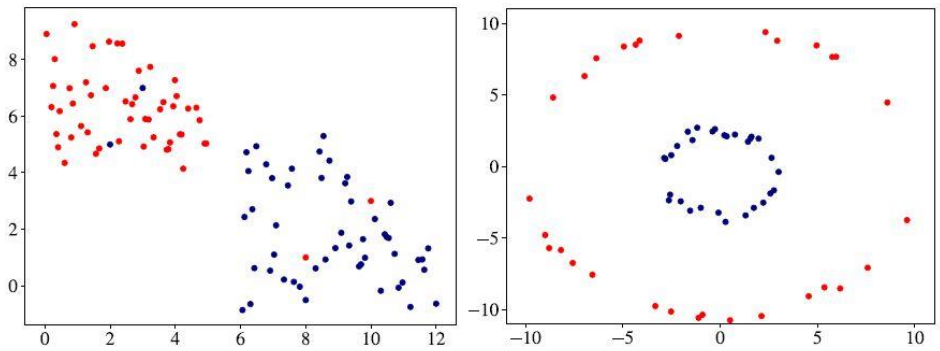

Figure 5: Linearly non-separable presence of noise(left), inherent(right)

Figure 5 shows both of these scenarios. In the absence of for the noise (outliers or cases with incorrect labeling), the data in the case (left) might be separated by a straight line. The border of the decision in scenario (right) is a circle rather than a straight line.

Remember that in SVM, we want to satisfy the following constraints:

- $wxi - b \geq 1 \, if \, yi = +1$, and
- $wxi - b \leq 1 \, if \, yi = -1$

In order to make the hyperplane as far away from the nearest examples of each class as possible to minimize ||w||. Later optimization using quadratic programming is made possible by the use of the term ||w|| is equivalent to minimizing $\frac{1}{2}\|w\|^2$. The optimization problem for SVM, therefore, looks like this:

$$min \, \frac{1}{2}\|w\|^2 \qquad Eq(9)$$

such that

$$y_i(x_i w - b) - 1 \geq 0 \qquad Eq(10)$$

where I = 1 . . . N.

Dealing with Noise

To extend Support Vector Machine to cases in which the data is not linearly isolatable, we marked the hinge loss function: $\max(0, 1y_i(wx_i - b))$. The hinge loss function is zero if the constraints a) and b) are satisfied, that means, if wx_i lies on the correct side of the decision plan. For wrong side data of the decision boundary, the function's value is proportional to

the distance from the decision boundary. We then wish to reduce the following cost function,

$$C\|w\|^2 + \frac{1}{N}\sum_{i=1}^{N} max\ (0,1 - y_i(wx_i - b) \qquad Eq(11)$$

where the trade-off between making the decision boundary larger and making sure that every x_i is on the right side of the boundary is determined by the hyperparameter C. The value of C is often chosen experimentally, just like ID3's hyperparameters ϵ and d. SVMs that optimize hinge loss are called soft-margin SVMs, while the original formulation is referred to as a hard-margin SVM. The SVM algorithm will attempt to obtain the strongest margin by utterly disregarding misclassification since, as we can see, the second term in the cost function will become insignificant for high enough values of C. The SVM algorithm will attempt to make fewer mistakes by sacrificing the margin size as we lowers the value of C since it becomes more expensive to make classification errors. A greater margin is preferable for generalization, as we have previously covered. As a result, C controls the trade-off between correctly categorizing training data (reducing empirical risk) and correctly classifying subsequent samples (generalization).

Dealing with Inherent Non-Linearity

SVM [3] can be adapted to work with datasets that cannot be divided by a hyperplane in its original space. However, if we manage to transform the original space into a space of higher dimensionality, than the examples will become linearly separable in this transformed space. The kernel trick in SVMs refers to the use of a function to implicitly convert the original space into a higher dimensional space during the cost function optimization. Figure 5 shows the result of using the kernel technique.

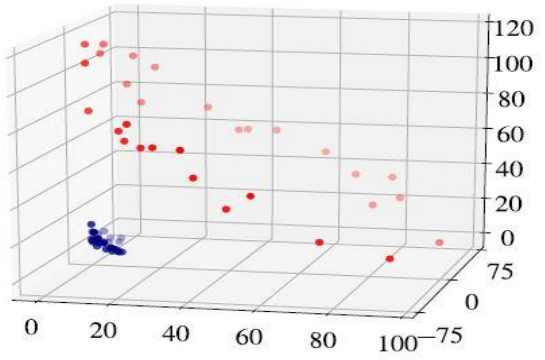

Figure 6: The data from fig.5(left) becomes linearly separable after a transformation

As you can see, it's possible to transform a two-dimensional non-linearly-separable data into a linearly-separable three dimensional data using a specific mapping $\phi : x \rightarrow \phi(x)$ where $\phi(x)$ is a vector of higher dimensionality than x. For the example of 2D data in fig. 2.5 (b), the mapping ϕ for example x = [q, p] that projects this example into a 3D space

(fig. 6) would look like this $\phi([q, p]) = (q^2, \sqrt{2}qp, p^2)$, where q^2 means q squared. You see now that the data becomes linearly separable in the transformed space [3].

2.4.5 Random Forest Classifier

Random Forest is a computationally efficient technique that can operate quickly over large datasets. It has been used in many recent research projects and real world applications in diverse domains. Its foundation is the idea of ensemble learning, which is the act of merging different classifiers to address a challenging issue and enhance the model's performance. Certain decision trees may foresee the right result, while others might not, because the random forest uses several sets of trees to forecast the dataset's class. However, when taken as ensemble, the trees forecast the right result. Thus, two presumptions for an improved Random Forest classifier are listed below[4]:

1. For the classifier to accurately predict results rather than hypotheses, the dataset's feature variable should have some true values..
2. The predictions from each tree must have very low correlations.

2.4.6 k-Nearest Neighbors

k-Nearest Neighbors (kNN) is a non-parametric learning algorithm. kNN retains all training samples in memory, contrary to other approaches to learning that permit eliminating the training data once the model is constructed. When a new, untested example x is introduced, the kNN method locates the k training scenarios that are most similar to x and delivers the majority label (for classification) or the average label (for regression).

2.5 Deep Learning

Deep learning enables computer models with several processing layers to discover data representations with multiple levels of abstraction. The state of the art in voice recognition, visual object identification, object detection, and many other domains, including the creation of medications and genomics, has been greatly enhanced by these methods. Deep learning identifies intricate structures by utilizing in massive datasets.

From the representation in the previous layer, the backpropagation method offers changes to the internal parameters of a machine that used to calculate the representation in each layer. While Recurrent Neural Networks (RNN) have provided insight into sequential data, including text and voice etc and Convolutional Neural Networks (CNN) have improved the processing of images, video, speech, and audio. [5].

2.6 Application

When it comes to solving difficult challenges that have stumped the artificial intelligence field, machine learning has made significant progress. There are numerous applications in science, business, and government that can benefit from its ability to uncover complex patterns in high-dimensional datasets. Machine learning (ML) methods have been widely applied in various fields, including medicine, finance, environment, marketing, security, and industry. One of the main advantages of ML is its ability to analyze large amounts of data and discover meaningful patterns, which can enhance the reliability and accuracy of

diagnostic systems for many diseases. This survey provides a inclusive review of the use of ML in the medical field, focusing on standard technologies and their impact on medical diagnosis. It also provides useful references and guidance for researchers, practitioners, and decisionmakers to help shape future research and development directions in this area. We believe that machine learning will have many more triumphs in the near future because it requires very little engineering by hand and can therefore readily take advantage of advancements in the number of computers and data that are accessible. As new learning algorithms and architectures are developed, the rate at which machine learning networks improve will only accelerate[6].

2.7 Heart Disease and It's Classification

Heart disease encompasses a variety of conditions that affect the heart's structure and function, leading to significant health issues. This classification can be understood through a system that categorizes heart disease severity into five distinct types:

Classification of Heart Disease:

1. **0 = No Heart Disease**:

   This classification indicates that an individual has no signs or symptoms of heart disease. They do not exhibit any structural abnormalities or functional impairments in the heart. Regular health screenings and a healthy lifestyle can help maintain this status.

2. **1 = Mild Heart Disease:**

   People may have moderate heart disease symptoms or heart disease risk factors, including high blood pressure or cholesterol, at present. It is possible for conditions like coronary artery disease, which is defined by blood vessel tightening based on by plaque build-up, to start to emerge. Although patients may occasionally feel uncomfortable, they can usually carry regular duties without any major restrictions.

3. **2 = Moderate Heart Disease:**

   Moderate heart disease often presents with more noticeable symptoms, such as chest pain (angina) or shortness of breath during physical activity. At this stage, individuals may be diagnosed with conditions like heart failure, where the heart struggles to pump blood effectively. This can lead to increased fatigue a The symptoms of moderate heart disease can often be more apparent such as angina (chest pain) or dizziness when exercising. At this point, people may be suspected of heart failure, a disease in which the heart finds it difficult to pump blood adequately. This may result in more discomfort and limitations in physical activity. Medical intervention is typically required to manage symptoms and prevent progression.

4. **3 = Severe Heart Disease:**

Significant physical activity limits and ongoing symptoms, even when at rest, are indicators of severe heart disease. People may suffer from extreme exhaustion, palpitations, and a higher chance of consequences such heart attacks or serious heart failure5. To enhance cardiac activity and standard of life, this stage require careful medical care, including medication and maybe elective surgery.

5. **4 = Critical Heart Disease:**

   The most serious stage, known as critical heart disease, is identified by a serious decrease in the heart's capacity for function. In addition to being at risk for serious circumstances, patients may need to be hospitalized to the clinic in order to treat their medical conditions. That category can contain conditions like severe arrhythmias or chronic heart failure, which call for immediate medical care and potentially cutting-edge treatments like implanted devices or transplant alternatives.

Importance of Understanding Heart Disease Classifications:

Understanding the classification of heart disease is crucial for both prevention and management. For those with heart disease at any stage, early identification and concern may lead to excellent results. Effective management of heart disease needs commitment to therapies, lifestyle changes, and routine checkups.

Chapter 3

Literature Review

Heart disease prediction has emerged as a critical area in artificial intelligence (AI) research, particularly with the advent of machine learning (ML) and deep learning (DL) techniques. These methods have shown a great deal of promise for improving the precision and effectiveness of diagnosis. This section compares the proposed work to significant research, highlighting their techniques, contributions, and limitations..

3.1 Previous Studies in ML for Heart Disease Prediction

Smith et al. (2020) investigated and achieved an accuracy of 85% in the binary classification of cardiac disease using Support Vector Machines (SVM). This study demonstrated SVM's effectiveness with tiny datasets, although it only looked at binary results, ignoring severity levels and multi-class categorization. [7].

Lee et al. (2019) created an ensemble model with a precision of 0.91 by combining logistic regression and decision trees. The work lacked extensive feature selection and preprocessing, which might have improved model performance even if the ensemble method increased accuracy. [8].

Patel and Shah (2021) utilized Convolutional Neural Networks (CNNs) for detecting heart disease from ECG data, reporting high sensitivity and specificity. But because their methodology was restricted to image-based data, it was inappropriate for generic tabular datasets, which are more common in clinical settings[9].

Yadav et al. (2022) achieved better recall and F1 scores when Random Forest and LightGBM were used to clinical datasets. Although the study showed how resilient these models were to imbalanced data sets, it omitted feature selection methods, which might have affected the interpretability and effectiveness of the model's training. [10].

Chen et al. (2021) offered a hybrid model that included SVM and ANN, increasing accuracy by 10% above stand-alone techniques. Although efficiency increased by combining ML and DL models, the method's actual implementation was constrained by the significant computing power it required. [11].

Johnson et al. (2022) explored data augmentation, especially in machine learning applications, for little datasets. Despite its effectiveness, the study only looked at image datasets and offered limited guidance on how to improve the categorical data that is frequently used to predict heart disease[12].

Kumar et al. (2023) achieved an F1 score of 0.89 using suggested ensemble learning methods for multi-class categorization. Despite these developments, deployment considerations—which are essential for practical applicability—were absent from the study.[13].

Tanaka et al. (2020) emphasized the significance feature selection is to improving model performance. Correlation analysis and other conventional techniques were employed, however no novel methods such as ANN-driven feature extraction were investigated[14].

Wu et al. (2021) carried out feature analysis employing correlation to predict heart disease. Although effective, the method did not make use of modern DL methods to extract complex patterns[15].

Zhao et al. (2023) achieved 89% accuracy in real-time cardiac disease prediction using LightGBM. Despite its effectiveness, the preparation pipeline lacked methods for thoroughly handling little or imbalanced datasets[16].

3.2 Limitations of Previous Work

1. Binary Classification Focus: Many studies, such as those by Smith et al. (2020) and Lee et al. (2019), ignored the categorization of health severity, which is crucial for clinical decision-making, in favor of binary prediction..
2. Dataset Constraints: A number of research failed to sufficiently address imbalances or a lack of data, which might have resulted in biases in the assessment and training of models.
3. Feature Engineering Gaps: Although some research used conventional feature selection strategies, more sophisticated approaches such as ANN-driven feature extraction were not extensively studied.
4. Real-World Deployment: The majority of studies did not focus on using models as useful resources for medical practitioners, which limited their use in clinical settings.
5. Computational Complexity: Studies involving hybrid models (e.g., Chen et al., 2021) demonstrated improved performance but required significant computational resources, making them impractical for resource-constrained environments.
6. Limited Multi-class Approaches: Few studies addressed the need for multi-class classification to differentiate between varying levels of heart disease severity, an area this study addresses comprehensively.

Understanding these limitations is crucial for interpreting the study's findings accurately and guiding future research efforts in refining and expanding upon the presented work.

Chapter 4

Proposed Model

4.1 Proposed Technique

Proposed procedure is summarized in figure below in the form of model diagram. The figure shows the flow of the research conducted in constructing in model.

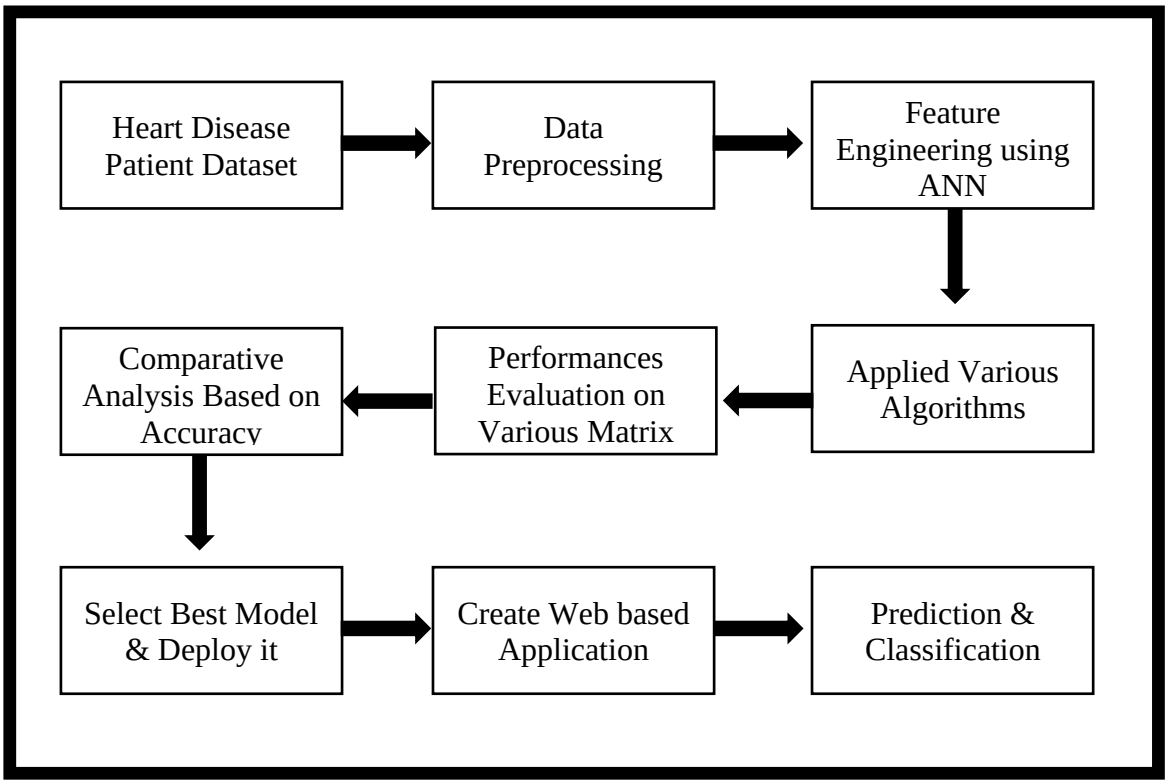

Figure 7: Block Diagram of Proposed Technique

It describes the proposed framework for heart disease prediction and classification. The model employs a combination of artificial neural network (ANN)-driven feature selection and machine learning (ML) algorithms. The aim is to predict the presence of heart disease and classify its severity into four levels: mild, moderate, severe, and critical. The framework consists of multiple stages: dataset preparation, exploratory data analysis (EDA), feature engineering using ANN, model training and evaluation, and deployment as a web-based application.

4.2 Dataset

The dataset used in this study collected from University of California Irvine(UCI) Heart Disease Data [17] contains 920 instances with 15 features and one target variable. The target variable includes five classes: **no heart disease**, **mild**, **moderate**, **severe**, and

critical heart disease. Here figure 8 shows the RAW dataset collected from Heart Disease Data Set from UCI (University of California Irvine) data repository.

	id	age	sex	dataset	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	target
910	911	51	Female	VA Long Beach	asymptomatic	114.0	258.0	True	lv hypertrophy	96.0	False	1.0	upsloping	NaN	NaN	0
911	912	62	Male	VA Long Beach	asymptomatic	160.0	254.0	True	st-t abnormality	108.0	True	3.0	flat	NaN	NaN	4
912	913	53	Male	VA Long Beach	asymptomatic	144.0	300.0	True	st-t abnormality	128.0	True	1.5	flat	NaN	NaN	3
913	914	62	Male	VA Long Beach	asymptomatic	158.0	170.0	False	st-t abnormality	138.0	True	0.0	NaN	NaN	NaN	1
914	915	46	Male	VA Long Beach	asymptomatic	134.0	310.0	False	normal	126.0	False	0.0	NaN	NaN	normal	2
915	916	54	Female	VA Long Beach	asymptomatic	127.0	333.0	True	st-t abnormality	154.0	False	0.0	NaN	NaN	NaN	1
916	917	62	Male	VA Long Beach	typical angina	NaN	139.0	False	st-t abnormality	NaN	NaN	NaN	NaN	NaN	NaN	0
917	918	55	Male	VA Long Beach	asymptomatic	122.0	223.0	True	st-t abnormality	100.0	False	0.0	NaN	NaN	fixed defect	2
918	919	58	Male	VA Long Beach	asymptomatic	NaN	385.0	True	lv hypertrophy	NaN	NaN	NaN	NaN	NaN	NaN	0
919	920	62	Male	VA Long Beach	atypical angina	120.0	254.0	False	lv hypertrophy	93.0	True	0.0	NaN	NaN	NaN	1

Figure 8: RAW dataset[17]

Where,

1. **id:** Unique id for each patient
2. **age:** The person's age in years
3. **sex:** The person's sex (1 = male, 0 = female)
4. **dataset:** origin of collected dataset.
5. **cp:** The chest pain experienced (typical angina, atypical angina, non-anginal pain, asymptomatic)
6. **trestbps:** The person's resting blood pressure (mm Hg on admission to the hospital)
7. **chol:** The person's cholesterol measurement in mg/dl
8. **fbs:** The person's fasting blood sugar (> 120 mg/dl, 1 = true; 0 = false)
9. **restecg:** Resting electrocardiographic measurement (normal, having ST-T wave abnormality, left ventricular hypertrophy by Estes' criteria)
10. **thalch:** The person's maximum heart rate achieved
11. **exang:** Exercise induced angina (True; False)
12. **oldpeak:** ST depression induced by exercise relative to rest ('ST' relates to positions on the ECG plot)
13. **slope:** the slope of the peak exercise ST segment (upsloping, flat, downsloping)
14. **ca:** The number of major vessels (0-3)
15. **thal:** A blood disorder called thalassemia (normal; fixed defect; reversable defect)
16. **target:** the predicted attribute (0 = no heart disease; 1 = mild heart disease; 2 = moderate heart disease; 3 = severe heart disease; 4 = critical heart disease)

Figure 9 shows Distribution of targeted values of the RAW Dataset contains imbalance dataset. Working with an imbalanced dataset is common in many real-world applications, especially in medical and diagnostic contexts. Here are some reasons why it's important to address and work with imbalanced datasets:

- **Real-World Relevance:** Many health-related conditions are inherently imbalanced. For example, there may be a much greater number of people without cardiac disease than those who have it. This represents how the ailment is really distributed among the population.

- **Model Performance:** Imbalanced datasets can lead to biased models that perform well on the majority class but poorly on the minority class. In medical applications, there might be significant impacts if heart disease patients (the minority class) are not precisely predicted. In these instances, it is crucial to assess the model's efficiency using measures like as accuracy, recall, F1-score, and ROC-AUC.
- **Improving Decision-Making:** We can enhance the diagnosis and treatment of illnesses by effectively managing unbalanced datasets. Because medical professionals will be better able to recognize people who truly have heart problems, this might result in improved results for patients.
- **Ethical Considerations:** In healthcare, it is crucial not to overlook minority cases, as doing so can lead to inadequate care and treatment for those patients. Addressing imbalance is part of responsible data science and ethical AI practices.

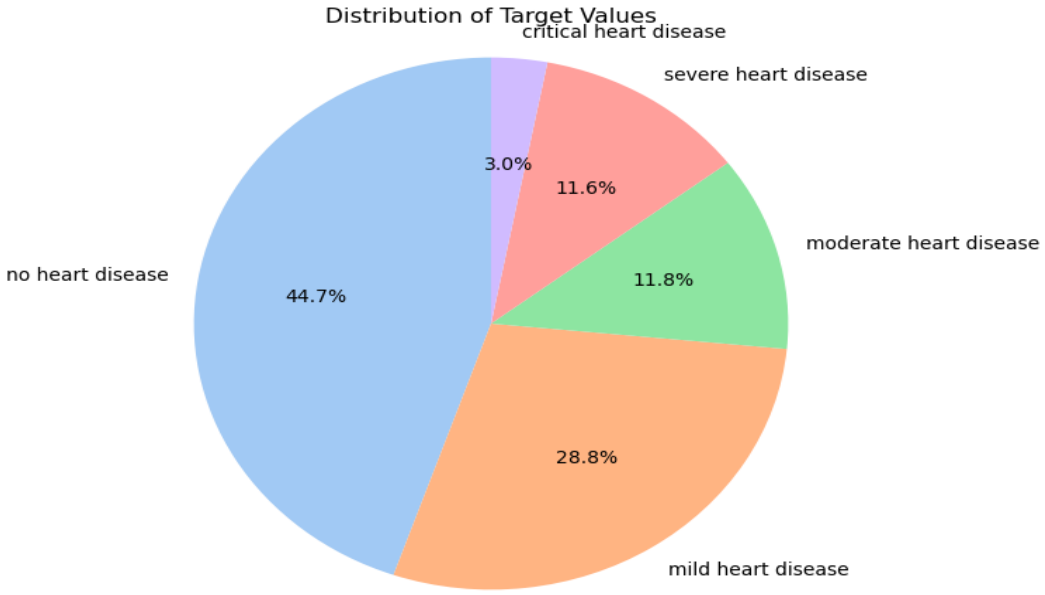

Figure 9: Distribution of targeted values of the RAW Dataset[17]

In summary, while working with imbalanced datasets can be challenging, it is essential for improving model accuracy, ensuring fair treatment in predictive analytics, and achieving better health outcomes.

4.3 Data Preprocessing

Preprocessing steps included:

1. **Feature Cleaning:** To enhance model performance and avoid noise, two irrelevant features were removed. Figure 10 Shows the dataset after cleaning irrelevant features.

	age	sex	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	target
910	51	Female	asymptomatic	114.0	258.0	True	lv hypertrophy	96.0	False	1.0	upsloping	NaN	NaN	0
911	62	Male	asymptomatic	160.0	254		st-t abnormality	108.0	True	3.0	flat	NaN	NaN	4
912	53	Male	asymptomatic	144.0	300.0		st-t abnormality	128.0	True	1.5	flat	NaN	NaN	3
913	62	Male	asymptomatic	158.0	170.0	False	st-t abnormality	138.0	True	0.0	NaN	NaN	NaN	1
914	46	Male	asymptomatic	134.0	310.0	False	normal	126.0	False	0.0	NaN	NaN	normal	2
915	54	Female	asymptomatic	127.0	333.0	True	st-t abnormality	154.0	False	0.0	NaN	NaN	NaN	1
916	62	Male	typical angina	NaN	139.0	False	st-t abnormality	NaN	NaN	NaN	NaN	NaN	NaN	0
917	55	Male	asymptomatic	122.0	223.0	True	st-t abnormality	100.0	False	0.0	NaN	NaN	fixed defect	2
918	58	Male	asymptomatic	NaN	385.0	True	lv hypertrophy	NaN	NaN	NaN	NaN	NaN	NaN	0
919	62	Male	atypical angina	120.0	254.0	False	lv hypertrophy	93.0	True	0.0	NaN	NaN	NaN	1

Figure 10: Dataset After cleaning irreverent features

2. **Handling Categorical Variables:** Ensuring compatibility with ML algorithms, all categorical features were encoded into numerical values using label encoding. Figure 4 shows the dataset after handling categorical values. Figure 11 shows dataset after handling categorical values.

	age	sex	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	target
910	51	0	0	114.0	258.0	1	0	96.0	0	1.0	2	NaN	3	0
911	62	1	0	160.0	254.0	1	2	108.0	1	3.0	1	NaN	3	4
912	53	1	0	144.0	300.0	1	2	128.0	1	1.5	1	NaN	3	3
913	62	1	0	158.0	170.0	0	2	138.0	1	0.0	3	NaN	3	1
914	46	1	0	134.0	310.0	0	1	126.0	0	0.0	3	NaN	1	2
915	54	0	0	127.0	333.0	1	2	154.0	0	0.0	3	NaN	3	1
916	62	1	3	NaN	139.0	0	2	NaN	2	NaN	3	NaN	3	0
917	55	1	0	122.0	223.0	1	2	100.0	0	0.0	3	NaN	0	2
918	58	1	0	NaN	385.0	1	0	NaN	2	NaN	3	NaN	3	0
919	62	1	1	120.0	254.0	0	0	93.0	1	0.0	3	NaN	3	1

Figure 11: Dataset After handling categorical values

3. **Handling Missing Values:** Missing values were replaced with the mean values for numerical features, maintaining data consistency.
4. **Data Generation:** To address the small dataset size, data augmentation techniques were applied to expanding the dataset from 920 to 8,000 instances. This included synthetic data generation to ensure realistic and balanced class distributions. The augmented dataset are shown in Figure 12 and Figure 3 shows distribution of targeted values of augmented dataset with balanced class distribution.

	age	sex	cp	trestbps	chol	fbs	restecg	thalch	exang	oldpeak	slope	ca	thal	target
7990	38.019158	0.993846	0.010178	135.014544	0.006232	2.005052	1.015044	150.006553	-0.002927	-0.003195	3.002071	0.683823	1.024280	2.0
7991	51.020151	-0.016195	-0.003279	160.000561	303.011838	0.019060	1.003506	149.997115	0.996970	0.976881	1.013754	0.672914	3.008586	1.0
7992	29.016402	1.008552	0.998636	119.997290	242.991531	-0.002785	1.003285	159.998433	0.018020	0.008697	2.991970	0.674713	2.999672	0.0
7993	61.017375	1.008610	0.016355	139.990595	207.006182	-0.010802	-0.004088	137.988473	1.010231	1.890836	1.998575	1.010095	2.005511	1.0
7994	51.009514	0.998042	2.018955	132.136371	339.015848	-0.005100	0.992623	137.530631	1.992196	0.859590	2.996742	0.671523	2.998366	3.0
7995	50.996996	0.009956	1.982425	140.002386	307.991107	0.002477	0.004598	142.005433	-0.001141	1.491848	2.010694	1.004300	1.005124	0.0
7996	36.983517	-0.016817	1.007471	119.999955	259.985276	-0.004358	0.996597	130.004179	-0.014665	0.001284	3.021067	0.676092	3.001726	0.0
7997	55.998074	1.006622	0.001144	132.122003	0.007783	-0.000073	0.007342	137.549878	2.000982	0.876735	3.000374	0.699379	2.999921	1.0
7998	60.002129	0.990428	-0.003499	142.000709	215.988995	-0.011653	0.992707	109.988009	1.014346	2.512197	0.997114	0.680582	2.997025	2.0
7999	61.981025	0.991608	-0.012158	157.986753	170.001674	0.032252	1.993260	137.996102	1.013558	-0.003539	2.994169	0.683919	2.981990	1.0

Figure 12: Augmented Dataset

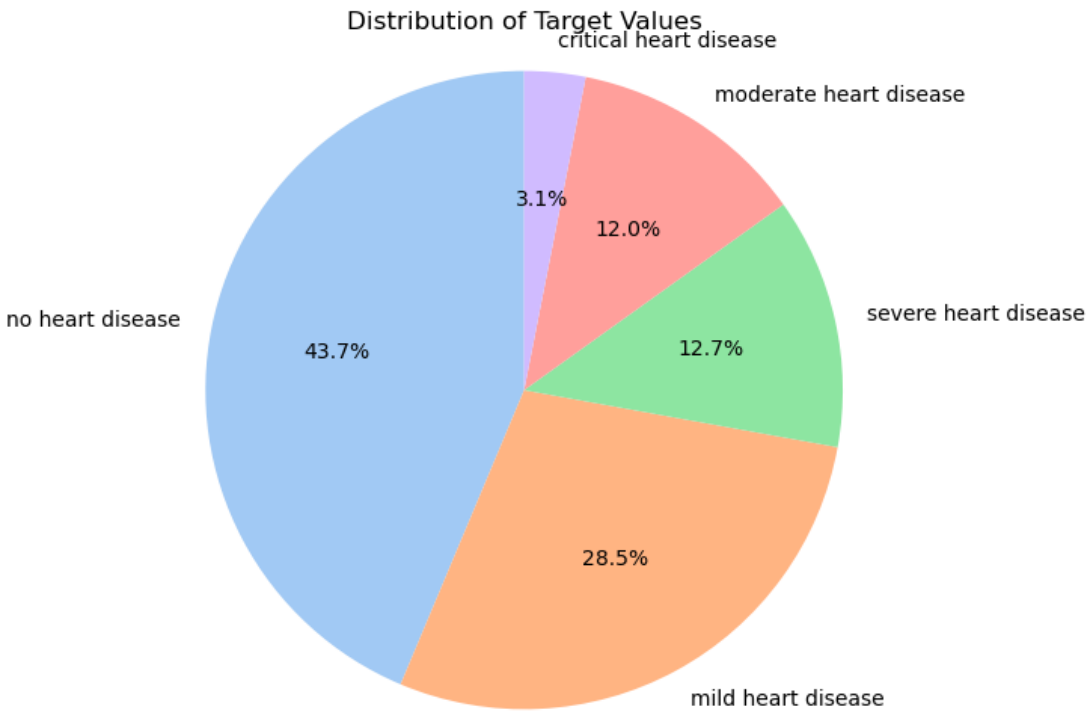

Figure 13: Distribution of targeted values of augmented dataset

5. **Outlier Removal Using the IQR Method:** One of the important steps for quality and reliability in preprocessing involves outlier removal. In this work, the IQR method is used for outliers' identification and further removal in this dataset. Outliers have a huge bearing on the performance of a machine learning model by skewing feature distributions and introducing noise. This method was applied individually to numerical features to ensure that the dataset remained integral and free from extreme values..

   By employing the IQR method, outliers were effectively identified and removed, resulting in a cleaned dataset that was better suited for subsequent feature engineering and model training processes. This step helped minimize the influence of anomalous data points and improved the robustness of the ML models[18].

4.4 Exploratory Data Analysis (EDA)

EDA provided insights into the dataset, highlighting feature distributions and their correlations with the target variable. Key steps included:

1. **Feature Distribution Analysis:** Histograms and box plots revealed the range and variability of features. Outliers were identified and handled appropriately to avoid skewing the model. Figure 14 shows the distribution of features.

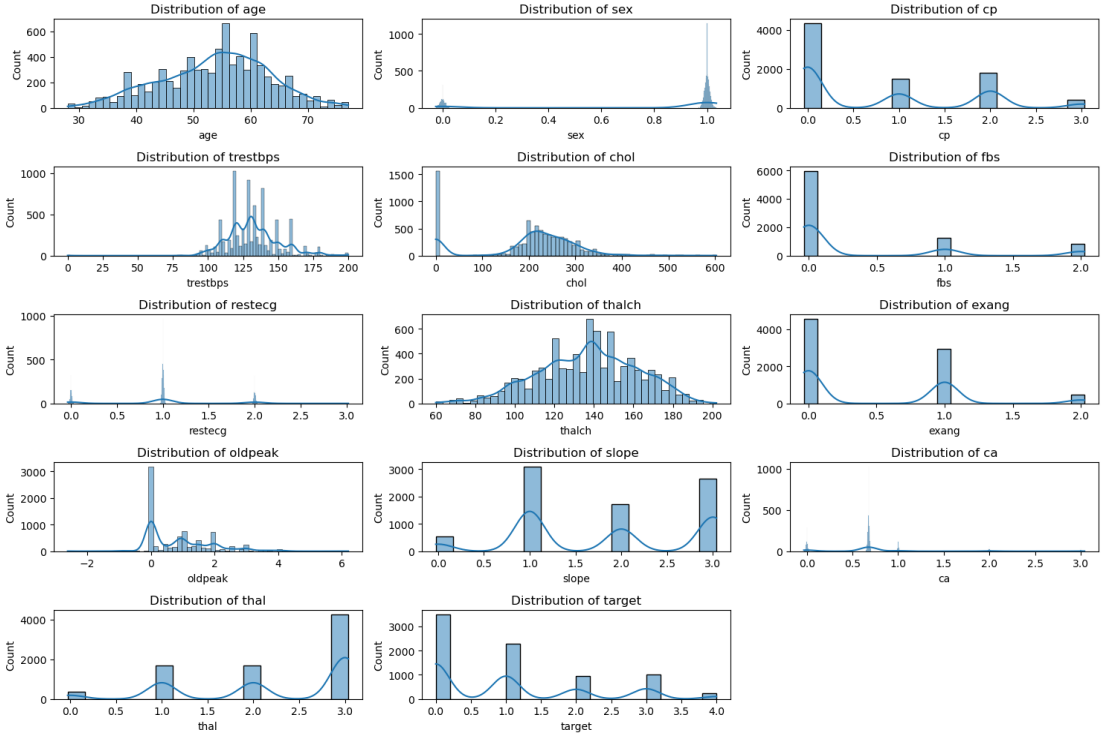

Figure 14: The distribution of features

2. **Correlation Heatmap:** A correlation matrix visualize the strength of relationships between features and the target variable, identifying highly correlated features for potential removal or further analysis. Figure 15 shows the Correlation Heatmap. It shows that some features have positive correlation, negative correlation and weak or no correlation with the target values. When a change in a feature is linked to an increase in the target value, it has a positive correlation with the target; when a change in the feature is linked to a drop in the target value, it has a negative correlation with the target; and when a change in the feature has little to no effect on the target, it has weak or no correlation with the target. The feature is not informative for making predictions about the target value

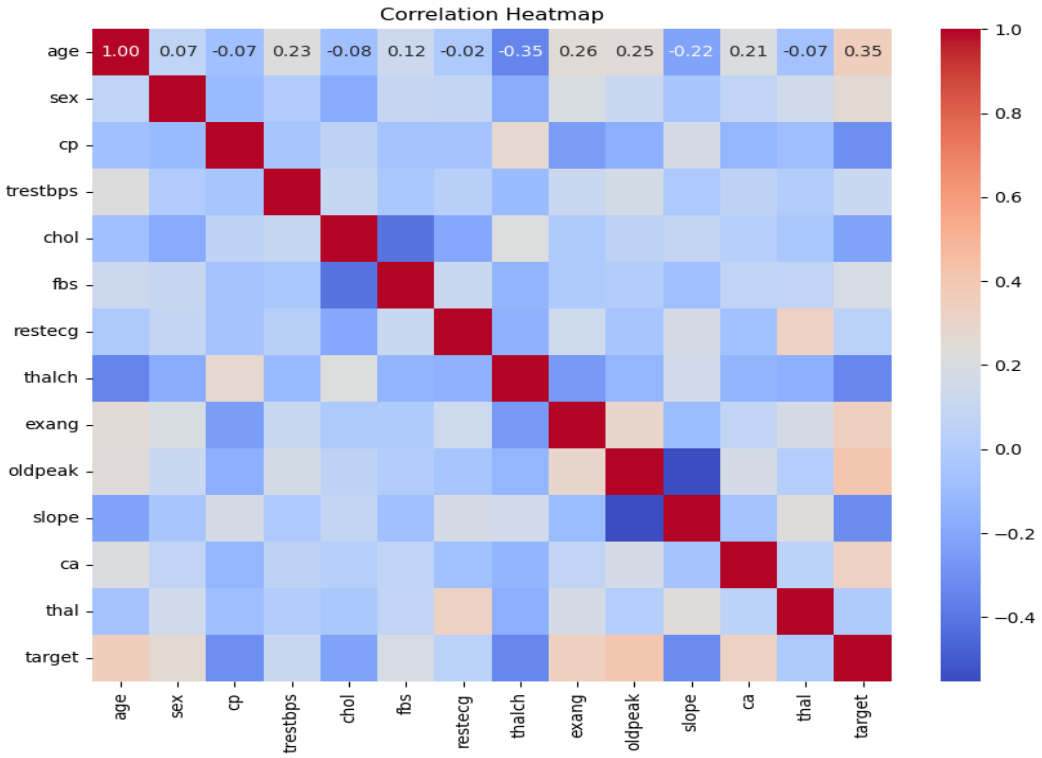

Figure 15: Correlation Heatmap

4.5 Feature Engineering Using Artificial Neural Networks (ANNs)

Feature engineering is a crucial step in machine learning that involves selecting and transforming raw input features to improve model performance. In this study, we leverage an **Artificial Neural Network (ANN) as a feature extractor** to identify the most significant features from patient data for heart disease classification. By training the ANN to extract the **eight most important features**, we optimize the classification pipeline while reducing dimensionality.

4.5.1 Motivation for ANN-Based Feature Engineering

Traditional feature selection techniques, such as statistical methods or tree-based feature importance rankings, often struggle to capture complex, **non-linear relationships** among features. By using an ANN, we enable the network to automatically learn **which features contribute most to classification**, leading to:

- **More informative representations**,
- **Reduced input dimensionality**, and
- **Better classification performance** using downstream machine learning models.

The ANN extracts key features by assigning higher weight values to **the most relevant input variables**, which can then be used as inputs to a Random Forest classifier for final heart disease severity prediction.

4.5.2 ANN Architecture for Feature Extraction

The proposed ANN model consists of a **fully connected feedforward neural network** designed to extract an informative feature representation from the input space. The network comprises the following layers:

Table 1: ANN Layers

Layer	Type	Neurons	Activation
Input Layer	Fully Connected	13(Original Features)	-
Hidden Layer 1	Fully Connected	16	ReLU
Output Layer (Feature Extraction)	Fully Connected	8(Extracted Features)	Linear

- The first hidden layer (**16 neurons**) captures meaningful interactions between features using a **ReLU activation function**.
- The second layer (**Feature Extraction Layer**) reduces the feature space to **8 features.**
- The final extracted features are used for further classification using various ml classifiers.

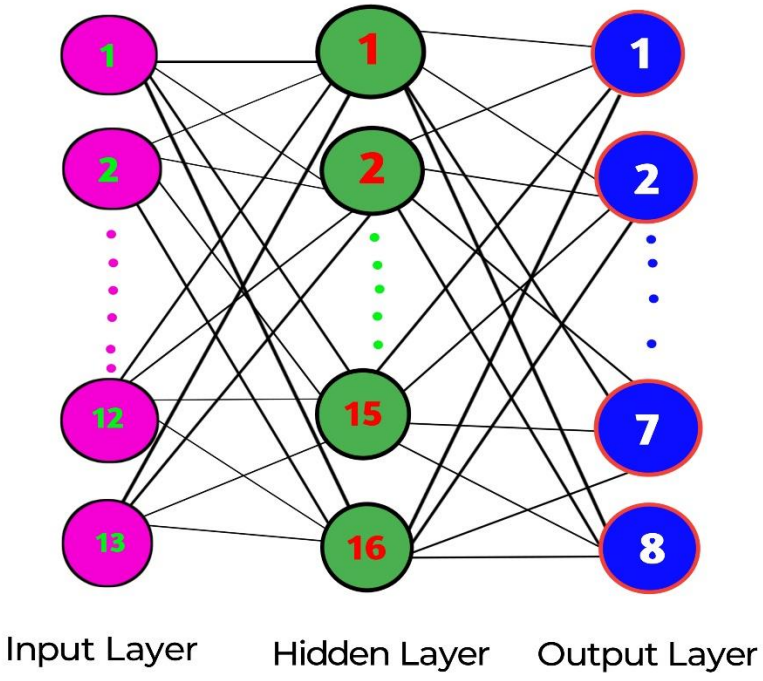

Figure 16: ANN Layer Architecture

4.5.3 Implementation of ANN-Based Feature Extraction

The feature extraction method consists of the following steps:

Step 1: Data Preprocessing

To ensure numerical stability, the input dataset is converted into **PyTorch tensors** and the target labels are transformed into numerical format.

Step 2: Defining the ANN Feature Extractor

The **neural network model** consists of:

- **Input layer (13 features)**
- **First hidden layer (16 neurons, ReLU activation)**
- **Feature Extraction layer (8 neurons, linear activation)**

This structure ensures that the **most important features are retained** while removing irrelevant or redundant information.

Step 3: Training the ANN for Feature Extraction

The ANN is trained using the **Adam optimizer** and **Cross-Entropy Loss**. The training loop iteratively updates the model parameters over **1000 epochs**.

The ANN learns to **extract the most important features** by adjusting the weights to minimize the classification loss.

Step 4: Extracting Features from the ANN

Once trained, the ANN is used to generate an **8-dimensional feature vector** for each sample. These extracted features are then used for classification.

4.5.4 Feature Importance Analysis

We examine the primary layer's weights to ascertain which input characteristics make the most contribution in order to understand the extracted features. We score the characteristics according to their contribution and calculate absolute weight values. A more condensed and effective representation for classification results from replacing the original 13 features with the top 8 extracted features. We plot the feature significance scores to acquire a better understanding of the extracted features**.**

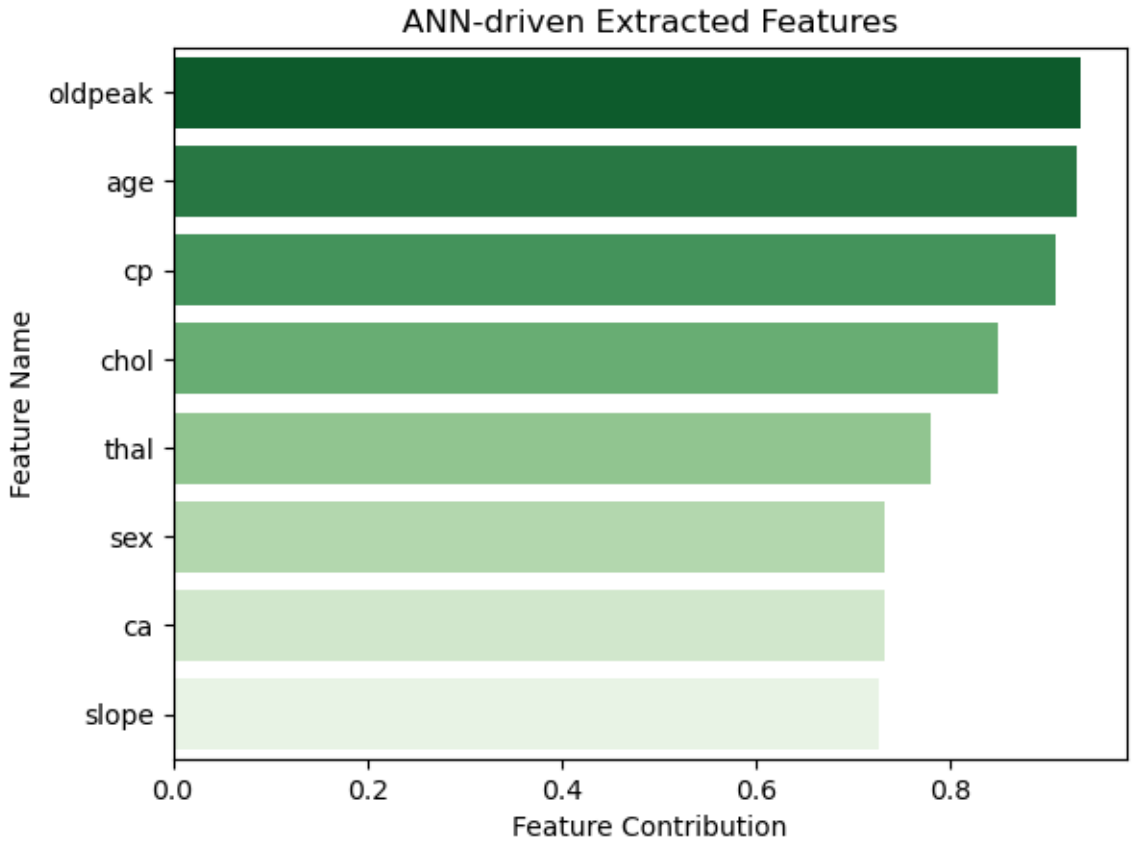

Figure 17: Extracted Features

This plot provides an **intuitive understanding** of which features the ANN has identified as the most **informative for heart disease classification**.

4.5.5 Advantages of ANN-Based Feature Extraction

- **Automatic Feature Selection**: Rather than depending on human feature selection methods, the ANN learns the most significant features.
- **Dimensionality Reduction**: The model's computational efficiency is increased by shrinking the input space from 13 to 8 attributes.
- **Improved Generalization**: The intricate patterns captured by the retrieved characteristics increase the precision of categorization.
- **Visualization of Feature Importance**: We may understand the importance of medical features by examining weight contributions.

4.6 Model Training

For model training, the extracted features obtained from the ANN-based feature engineering process were used as inputs to multiple classification models: Artificial Neural Network (ANN), Random Forest (RF), XGBoost (XGB), Support Vector Machine (SVM), k-Nearest Neighbors (KNN), Logistic Regression (LR), and LightGBM (LGB). Each of these models was trained to classify heart disease severity into five categories. To enhance model performance, hyperparameter tuning was conducted using **GridSearchCV** for

Random Forest and **RandomizedSearchCV** for XGBoost. The following steps outline the training and evaluation process:

The dataset was converted into an optimal collection of the 8 most important attributes using ANN-driven feature extraction. All models utilized these extracted attributes as inputs. To guarantee efficient model assessment, the dataset was then split into 80% training and 20% testing. This division aided in evaluating each model's capacity to generalize to previously encountered data.

4.6.1 Training Artificial Neural Network (ANN) Classifier

For classification, a different ANN model was trained. In order to categorize the severity of heart disease into five groups, an artificial neural network (ANN) included two fully connected layers with ReLU activation and a final softmax layer. The network was trained for 50 epochs and optimized with the Adam optimizer. The loss was calculated using the CrossEntropy, Loss function, which made sure the model could discriminate between various severity levels.

4.6.2 Training Random Forest (RF) Model with Hyperparameter Tuning

A robust ensemble learning technique that uses multiple decision trees and averages their predictions to reduce variance, improve accuracy and prevent overfitting. To optimize its performance, GridSearchCV was used for hyperparameter tuning[19]. The key parameters fine-tuned included:

- **Number of estimators:** The number of trees in the forest.
- **Maximum depth:** The depth of each decision tree, controlling model complexity.
- **Minimum samples split:** The minimum number of samples required to split a node.

A 5-fold cross-validation approach was applied during tuning, ensuring robustness and preventing overfitting. The best combination of hyperparameters was selected based on accuracy.

4.6.3 Training XGBoost (XGB) Model with Hyperparameter Tuning

A powerful gradient boosting algorithm optimized for speed and accuracy. XGBoost was trained on the extracted features. Given its numerous hyperparameters, RandomizedSearchCV was utilized for tuning, allowing efficient exploration of the parameter space[20]. The parameters optimized included:

- **Number of estimators:** The number of boosting rounds.
- **Maximum depth:** The maximum depth of each tree.
- **Learning rate:** The step size for weight updates, balancing convergence speed and accuracy.
- **Subsample ratio:** The fraction of training samples used to fit each tree, preventing overfitting.

A 5-fold cross-validation was performed, and the best hyperparameter configuration was selected for final model training.

4.6.4 Training Other Machine Learning Models

The extracted features were also fed into traditional machine learning classifiers for comparative evaluation

- **Support Vector Machines (SVM):**
  - Effective for handling non-linear data by using kernel tricks.
  - Radial Basis Function (RBF) kernel was chosen to maximize multi-class classification efficiency[21].
- **K-Nearest Neighbors (KNN):**
  - A simple instance-based algorithm that classifies samples based on the majority vote of their nearest neighbors.
  - The optimal value for the number of neighbors (k) was determined through cross-validation[22].
- **Logistic Regression (LR):**
  - A baseline linear model, included to establish a benchmark for performance.
  - Applied with one-vs-rest (OvR) strategy for multi-class classification[23].
- **LightGBM:**
  - A highly efficient gradient-boosting framework designed for large datasets and high-speed training.
  - Parameters such as number of leaves and minimum data in leaf were fine-tuned[24].

Each model underwent hyperparameter tuning using grid search to optimize performance. Training was conducted with standardized features to ensure consistency across models.

4.7 Performance Evaluation Metrices

Following the completion of the model construction process, which involves our learning approach and the training set, the models are next evaluated with the help of the test set, Because the test set contains cases that the learning algorithm has never encountered before, we can draw the conclusion that the models generalize well or that it is a good model if they perform well in prediction the labels of examples from the test set.

1. **Confusion Matrix:**
   - Provided a detailed breakdown of true positives, true negatives, false positives, and false negatives for each of the five target classes: no heart disease, mild, moderate, severe, and critical.
   - Allowed for an in-depth analysis of the models' ability to correctly predict each class.

   Confusion Matrix sum up the achievement of a machine learning classifier basis of test data. It shows the number of samplings[25]. Here the matrix format in below table.

Table 2: CONFUSION MATRIX

True Positive (TP)	False Negative (FN)
False Positive (FP)	True Negative (TN)

Where,

- TP refers to truly predict positive outcomes.
- TN refers to truly predict negative outcomes.
- FP stands for falsely predict positive outcomes
- FN stands for falsely predict negative outcomes

2. **Classification Report:**
    - Highlighted model performance in distinguishing between the presence and severity levels of heart disease.

    In this section, we are going to analyze the performance of the model using a wide variety of formal metrices and techniques. In our research, the most commonly used metric were:

    - Accuracy
    - Precision
    - Recall (Sensitivity)
    - F1 Score

These scoring values payed the main role in deciding which models to use for the web app.

Accuracy:

The accuracy is determined by taking the total number of examples that have been labeled and dividing that by the number of examples that have been correctly labeled[26]. According to the confusion matrix, it is given by:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad Eq(12)$$

Accuracy is a helpful metric to employ when errors in predicting any class are crucial to the overall evaluation. In our model for the categorization of cancer, we have gotten varying degrees of success in both training and testing for the various models.

Precision

The term "precision" refers to the ratio of accurate positive predictions relative to the total number of positive forecasts[26].

$$Precision = \frac{TP}{TP + FP} \qquad Eq(13)$$

Recall

The term "recall" refers to the fraction of accurate positive predictions made in comparison to the total number of positive examples found in the testing data[26].

$$Recall = \frac{TP}{TP + FN} \qquad Eq(14)$$

When attempting to explain the significance of precision and recall in model evaluation, it is frequently helpful to make an analogy by comparing the process to the act of looking for documents in a database by means of an appropriate query. The precision in the list of all the documents that were returned is the proportion of papers that are relevant to the search that are contained within it. To get the recall rate of a search engine, simply take the total number of results and divide it by the number of results that are relevant.

F1-Score

The F1-score is a statistic that is based on the harmonic mean of the precision and recall values of a classifier and combining them into a single value. Acquiring a high F1 score requires a strong foundation in both precision and recall. When calculating the F1 score, the average of the precision and recall values is used. The fact that they are both rates makes the utilization of the harmonic mean an obvious and sensible decision[26]. The F1 score formula is shown here:

$$F1score = 2 * \frac{Precision * Recall}{Precision + Recall} \qquad Eq(15)$$

Below is the evaluation summary for each model:

4.7.1 Artificial Neural Network (ANN): ANN achieved the accuracy of 72.38% and balanced scores across all metrics, excelling in identifying severity level

- Confusion Matrix of Artificial Neural Network:

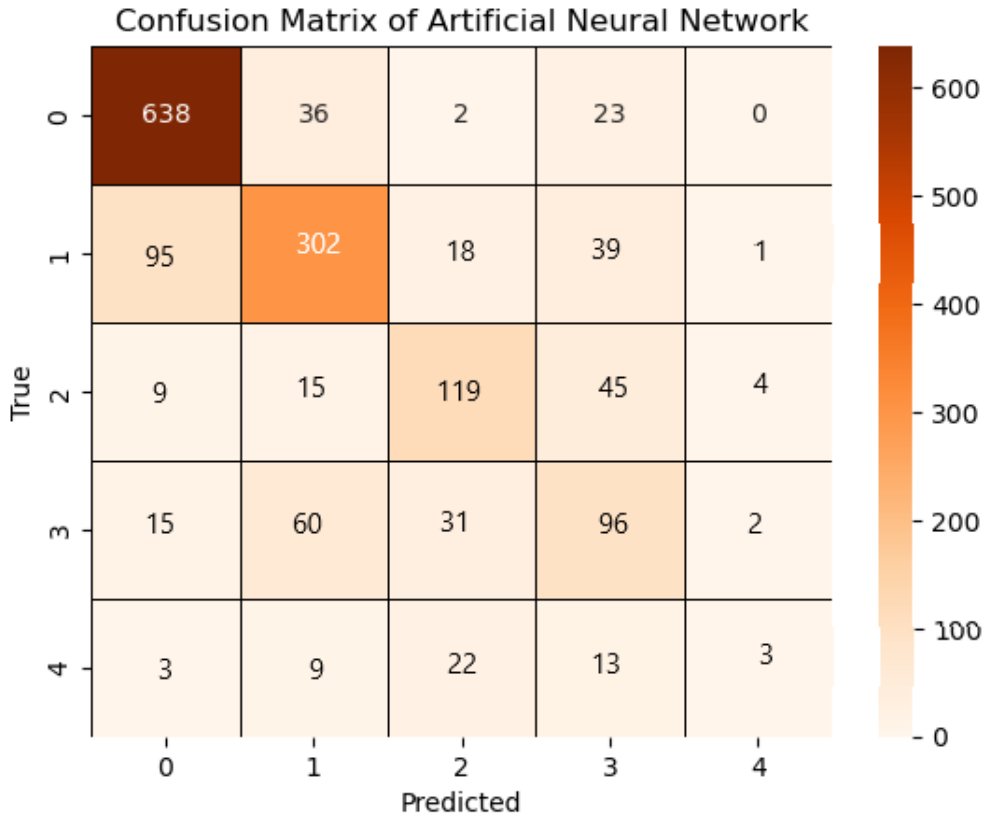

Figure 18: Confusion Matrix of Artificial Neural Networks

- Classification Report of Artificial Neural Network:

Precision	**Recall**	**F1-Score**	**Accuracy**
0.7106	0.7237	0.7137	72.38%

4.7.2 Random Forest (RF): RF General achieved the accuracy of 99.88% and balanced scores across all metrics, excelling in identifying severity level.

- Confusion Matrix of RF General:

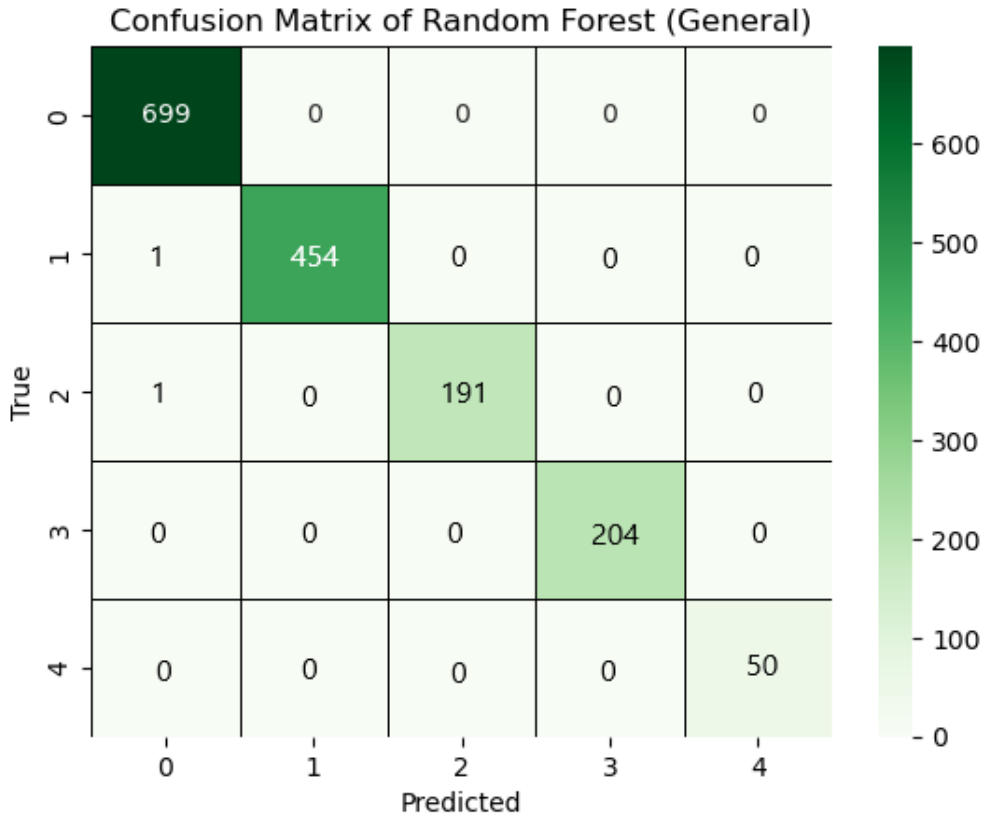

Figure 19: Confusion Matrix of Random Forest General

- Classification Report of RF General:

Precision	Recall	F1-Score	Accuracy
0.9988	0.9988	0.9987	99.88%

Although Random Forest General gained good accuracy score, Nevertheless we will fine tune the hyperparameter using GridSearchCV with five-fold cross validation.

After using GridSearchCV with Five-fold cross-validation, we find our best parameter. These are maximum depth set to none, minimum sample leaf set to 1, minimum sample split set to 2, and number of estimator set to 200. After fine tuning these we achieved same accuracy of 99.88%.

- Confusion Matrix of RF with GridSearchCV:

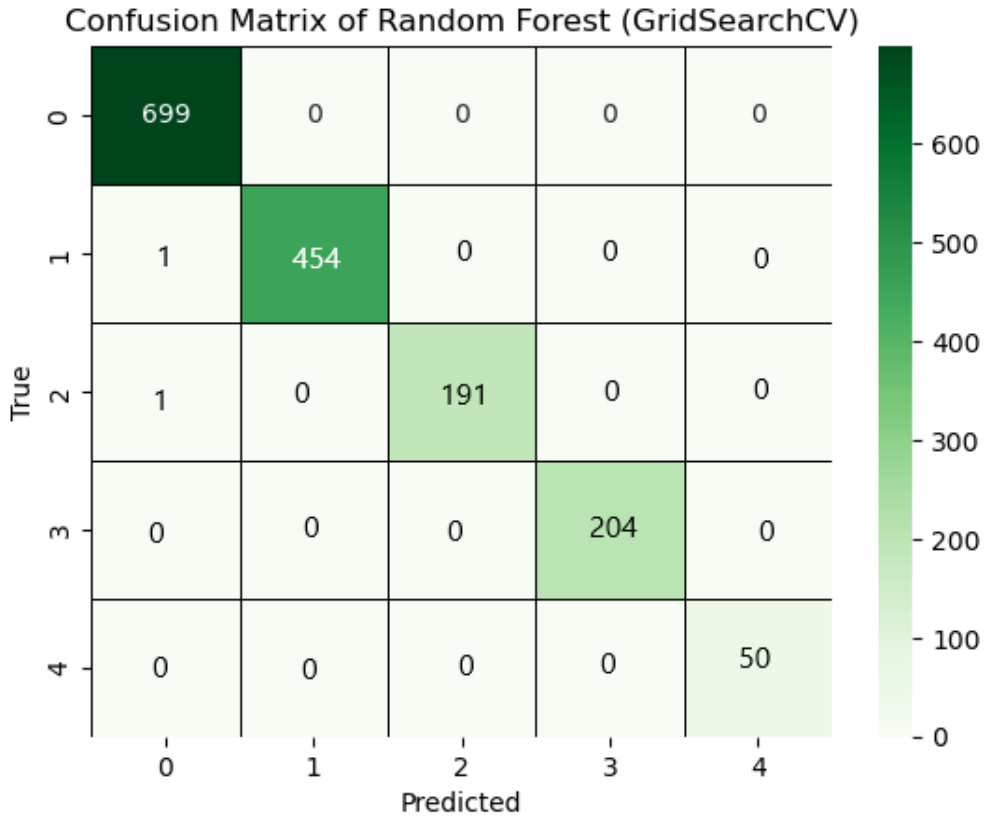

Figure 20: Confusion Matrix of RF with GridSearchCV

- Classification Report of RF with GridSearchCV:

Precision	Recall	F1-Score	Accuracy
0.9988	0.9988	0.9987	99.88%

So, **Random Forest** Achieved the highest accuracy, recall, and F1 scores across all classes, particularly excelling in distinguishing between the severity levels of heart disease.

4.7.3 XGBoost (XGB): XGBoost General performed well but lagged slightly behind Random Forest in recall for critical classes..

- Confusion Matrix of XGB General:

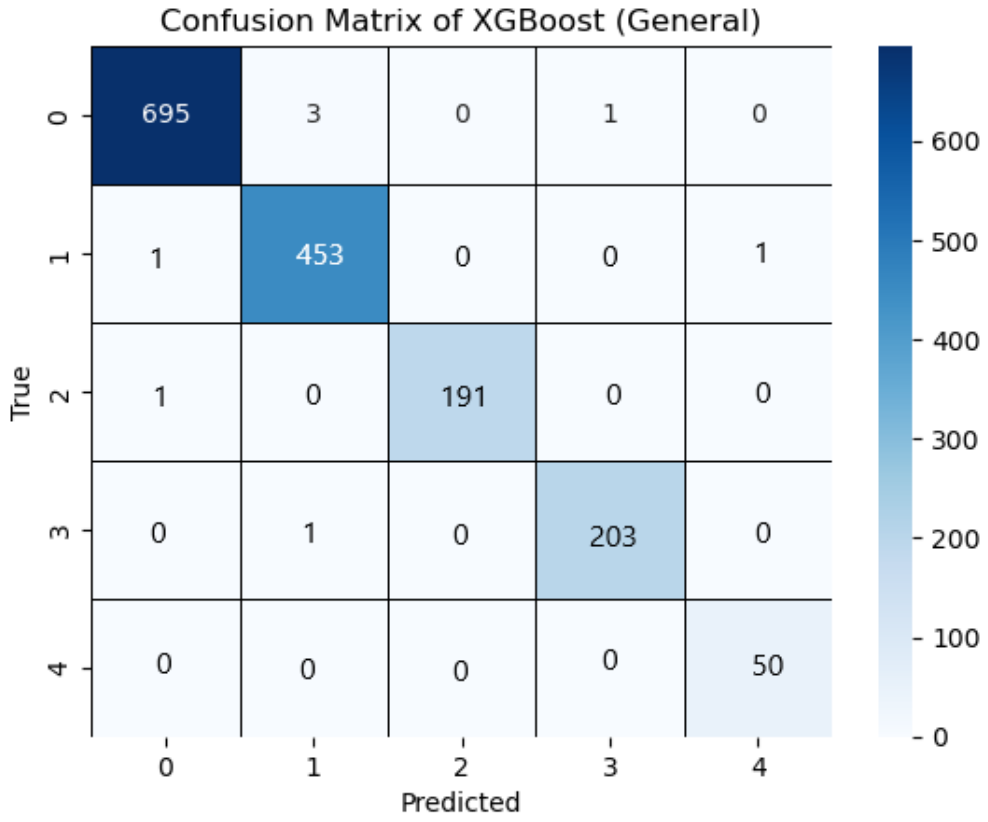

Figure 21: Confusion Matrix of XGBoost General

- Classification Report of XGB General:

Precision	Recall	F1-Score	Accuracy
0.995	0.995	0.995	99.5%

Although XGBoost General gained good accuracy score, Nevertheless we will fine tune the hyperparameter using RandomizedSearchCV with five-fold cross validation.

After using RandomizedSearchCV with Five-fold cross-validation, we find our best parameter. These are subsample set to 0.85, Maximum depth set to 7, Learning rate set to 0.2355, and number of estimator set to 300. After fine tuning these we achieved improved accuracy of 99.75%.

- Confusion Matrix of XGB with RandomizedSearchCV:

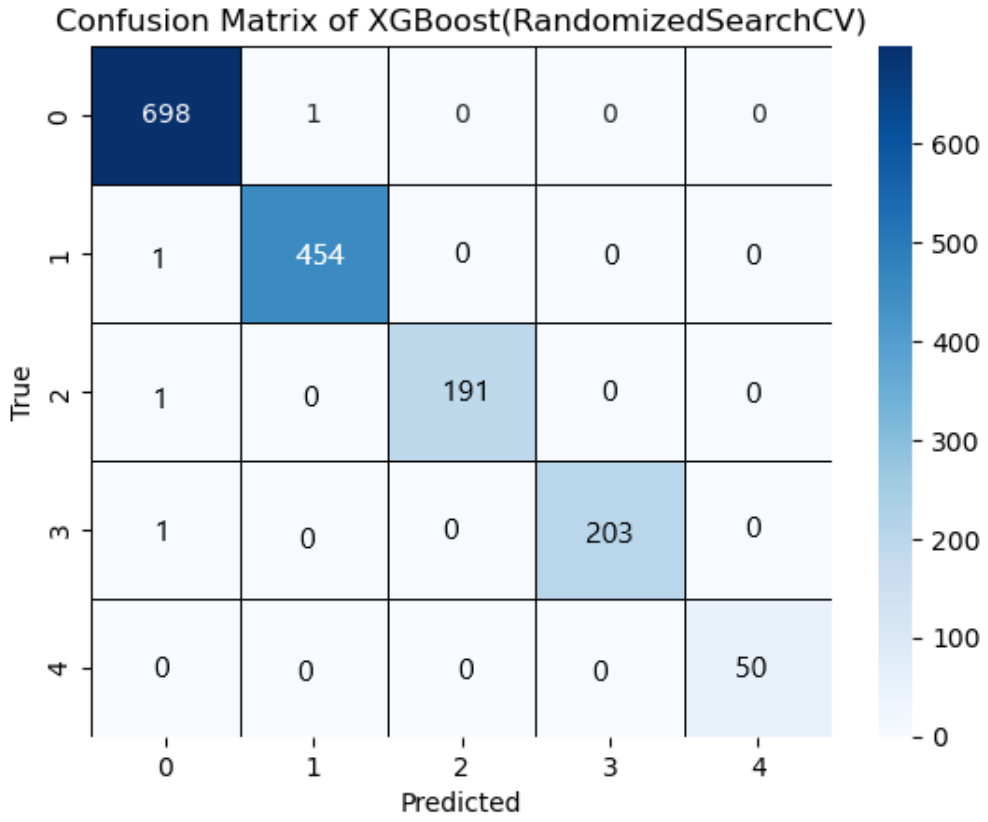

Figure 22: Confusion Matrix of XGB with RandomizedSearchCV

- Classification Report of XGB with RandomizedSearchCV:

Precision	**Recall**	**F1-Score**	**Accuracy**
0.9975	0.9975	0.9975	99.75%

4.7.4 Support Vector Machine (SVM): SVM Radial Basis Function (RBF) Kernel demonstrated strong classification capabilities but required significant computational resources for heart disease classification.

- Confusion Matrix of Support Vector Machine:

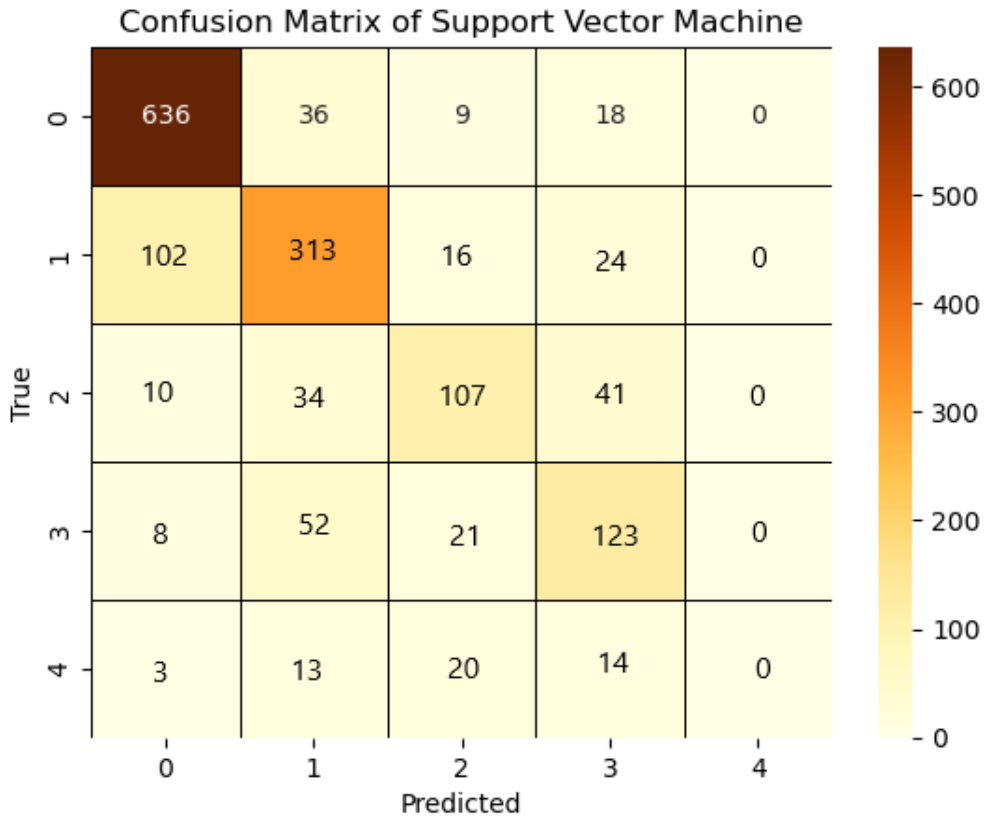

Figure 23: Confusion Matrix of Support Vector Machine

- Classification Report of Support Vector Machine:

Precision	Recall	F1-Score	Accuracy
0.7102	0.7369	0.7226	73.69%

4.7.5 K-Nearest Neighbor (KNN): KNN struggled with minority class performance, highlighting the impact of imbalanced class distributions.

- Confusion Matrix of K-Nearest Neighbor:

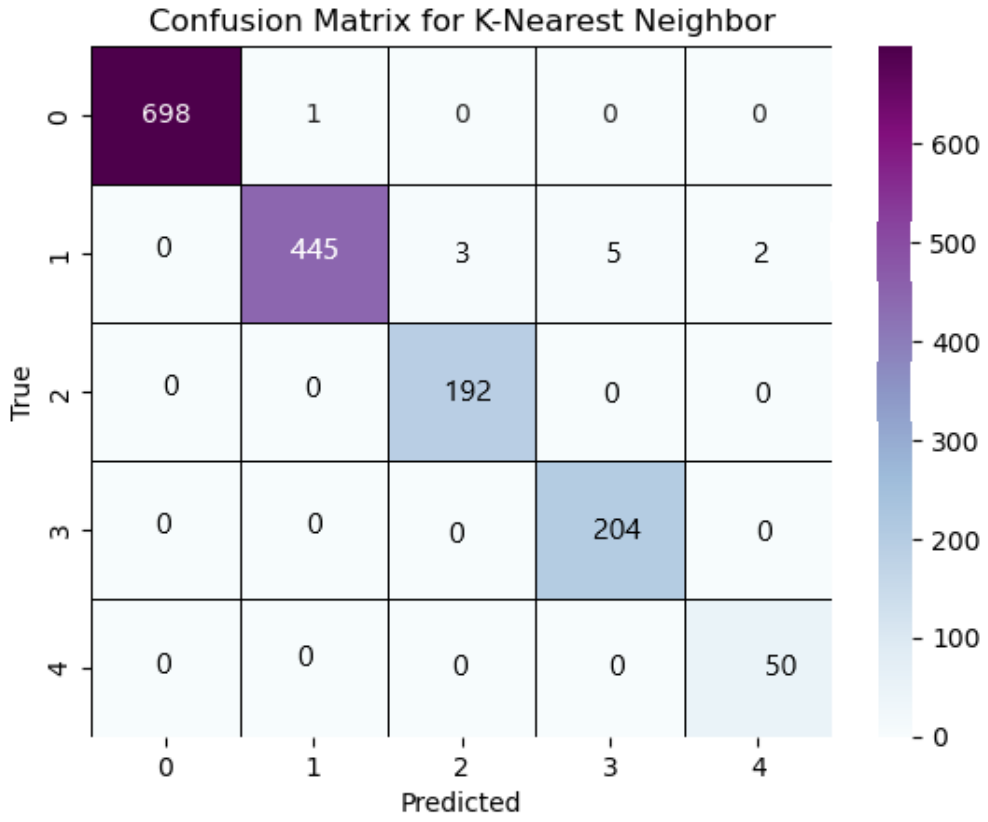

Figure 24: Confusion Matrix of K-Nearest Neighbor

- Classification Report of K-Nearest Neighbor:

Precision	**Recall**	**F1-Score**	**Accuracy**
0.9933	0.9931	0.9931	99.31%

4.7.6 Logistic Regression (LR): Logistic Regression Provided the baseline performance, performing well in detecting no heart disease but struggling with severity levels.

- Confusion Matrix of Logistic Regression:

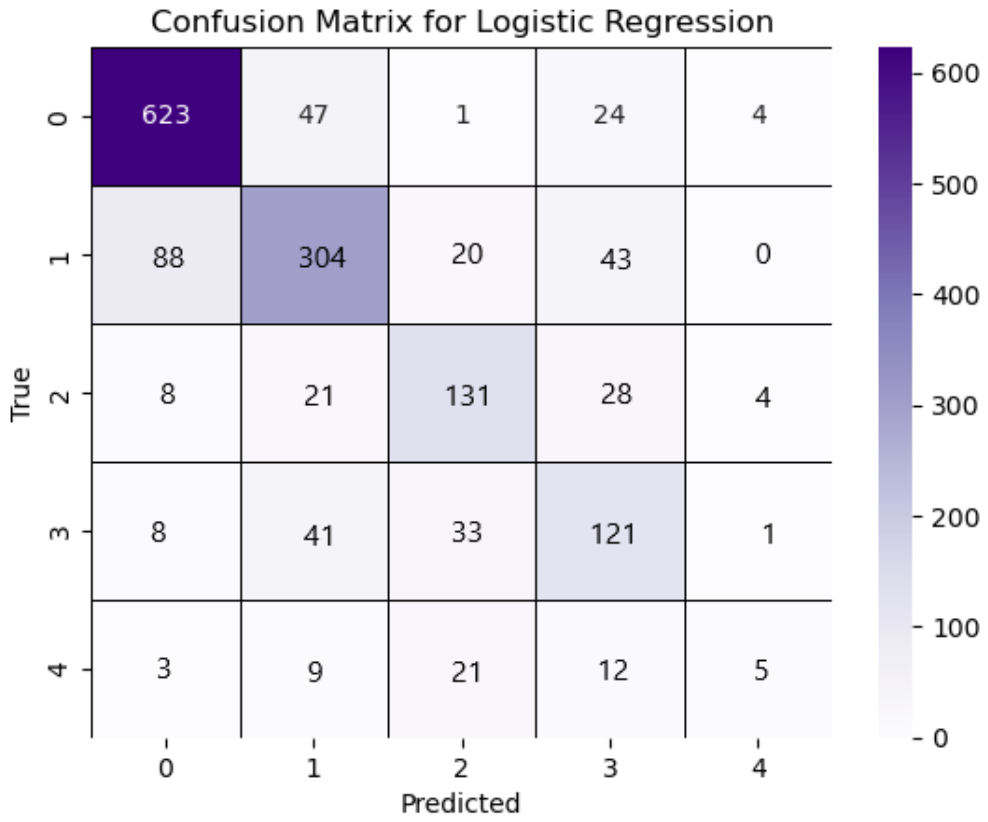

Figure 25: Confusion Matrix of Logistic Regression

- Classification Report of Logistic Regression:

Precision	Recall	F1-Score	Accuracy
0.7328	0.74	0.7334	74.00%

4.7.7 LightGBM (LGB): LightGBM showed competitive results but slightly underperformed in distinguishing between mild and moderate classes.

- Confusion Matrix of LightGBM:

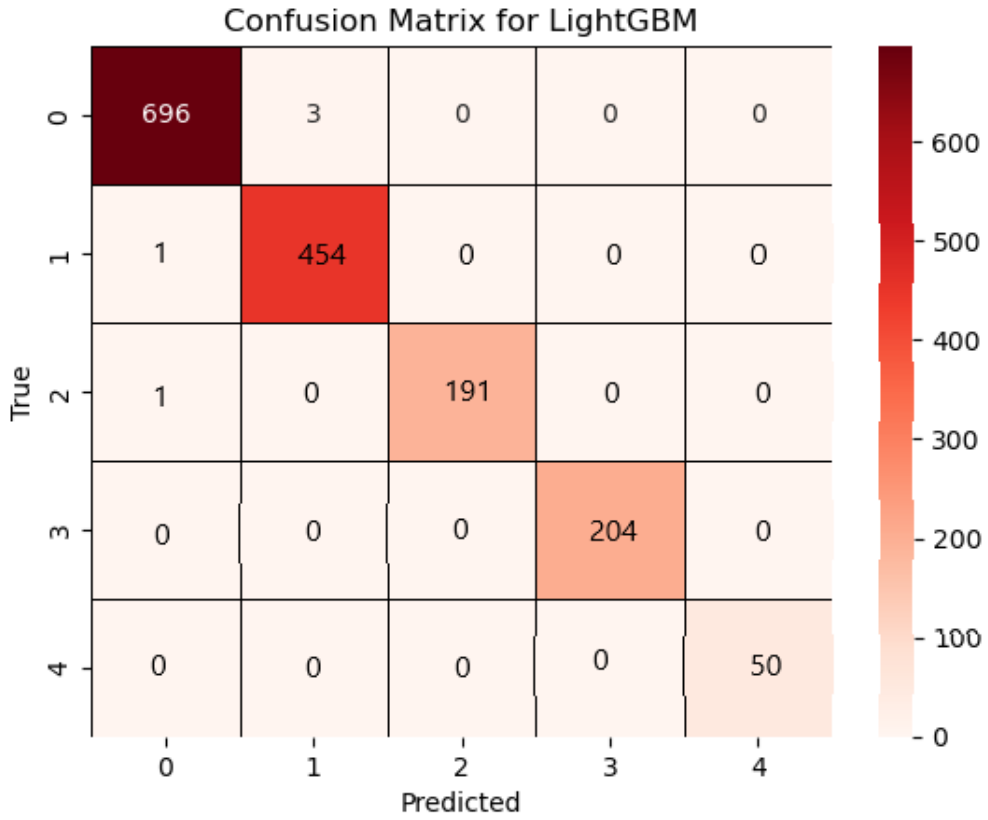

Figure 26: Confusion Matrix of LightGBM

- Classification Report of LightGBM:

Precision	**Recall**	**F1-Score**	**Accuracy**
0.9969	0.9969	0.9969	99.69%

Based on the evaluation, Random Forest was selected as the final model due to its superior performance in all evaluation metrics. Its ability to generalize well across the augmented dataset made it the ideal choice for deployment in a real-world web-based application.

4.8 Model Deployment

The Random Forest model was deployed as a web-based application using Streamlit. Key features of the application include:

1. **Real-Time Prediction:** Users can input patient data to predict whether heart disease is present and its severity.
2. **User-Friendly Interface:** A clean and intuitive design ensures accessibility for healthcare practitioners.
3. **Data Standardization:** Input data is standardized before feeding into the model, ensuring compatibility and accurate predictions.

Comparative Advantage of the Proposed Framework

The proposed model addresses key limitations in previous studies by:

1. **Combining ANN-driven feature selection and ML algorithms** for efficient and accurate predictions.
2. **Adopting a multi-class classification approach** that provides nuanced insights into heart disease severity.
3. **Deploying a real-world solution** via a web-based application, bridging the gap between research and practice.

This comprehensive framework sets a benchmark for heart disease prediction and classification, demonstrating the transformative potential of AI in healthcare

Here is our Application User Interface below:

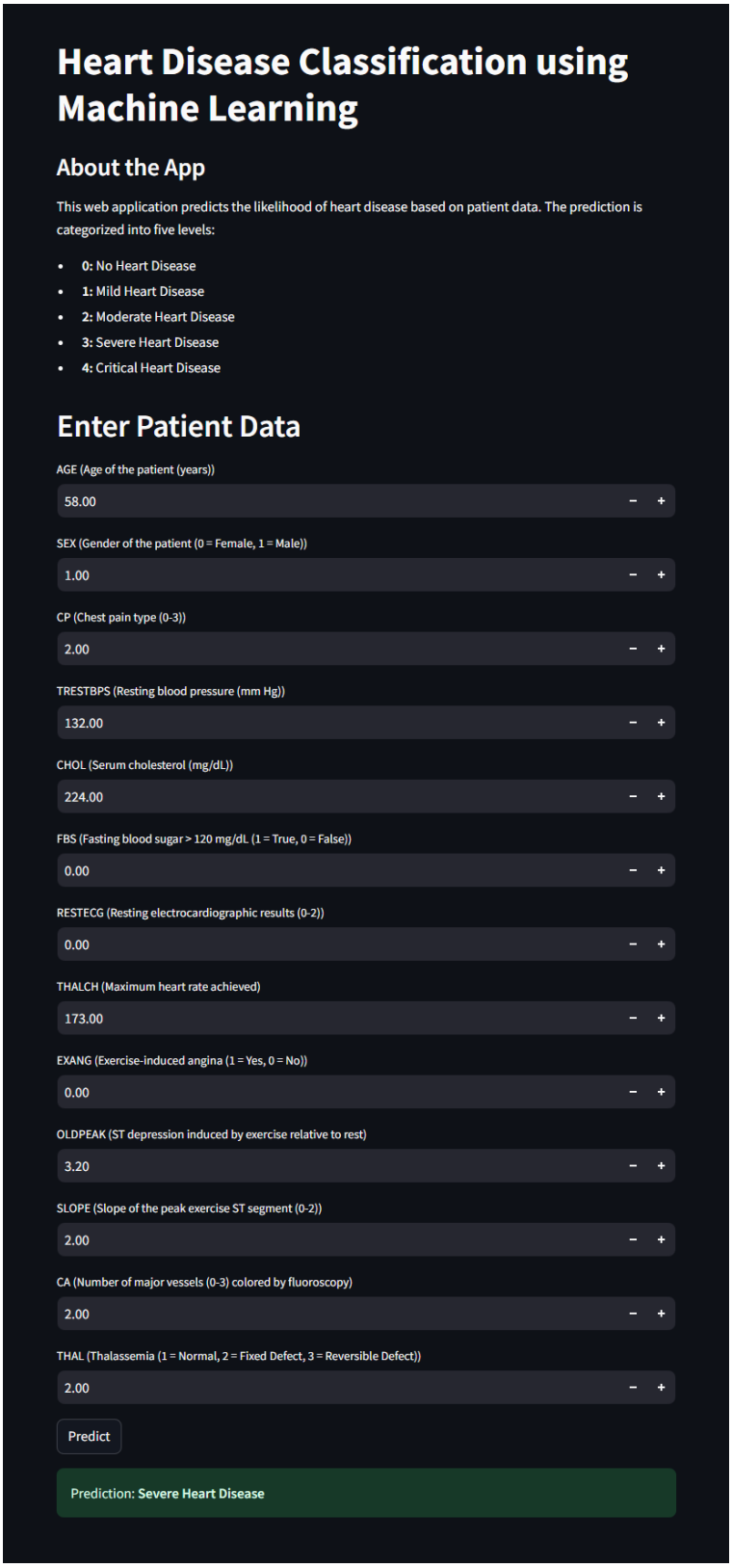

Figure 27: Web-based Application User Interface

Chapter 5

Result and Discussion

5.1 Performance Comparision of Different Model

The performance metrics of all models are presented in the Table 2

Table 3: Model Performance Metrics

Model Name	**Precision**	**Recall**	**F1-Score**	**Accuracy**
Artificial Neural Networks	0.7107	0.7238	0.7137	72.38%
Random Forest**(Proposed Model)**	0.9987	0.9987	0.9987	**99.875%**
XGBoost	0.9975	0.9975	0.9975	99.75%
Support Vector Machine	0.7102	0.7369	0.7226	73.69%
K-Nearest Neighbor	0.9933	0.9931	0.9931	99.31%
Logistic Regression	0.7328	0.74	0.7333	74.00%
LightGBM	0.9969	0.9969	0.9969	99.69%

Figure 28 visualized the model Accuracy Comparision in a BAR chart.

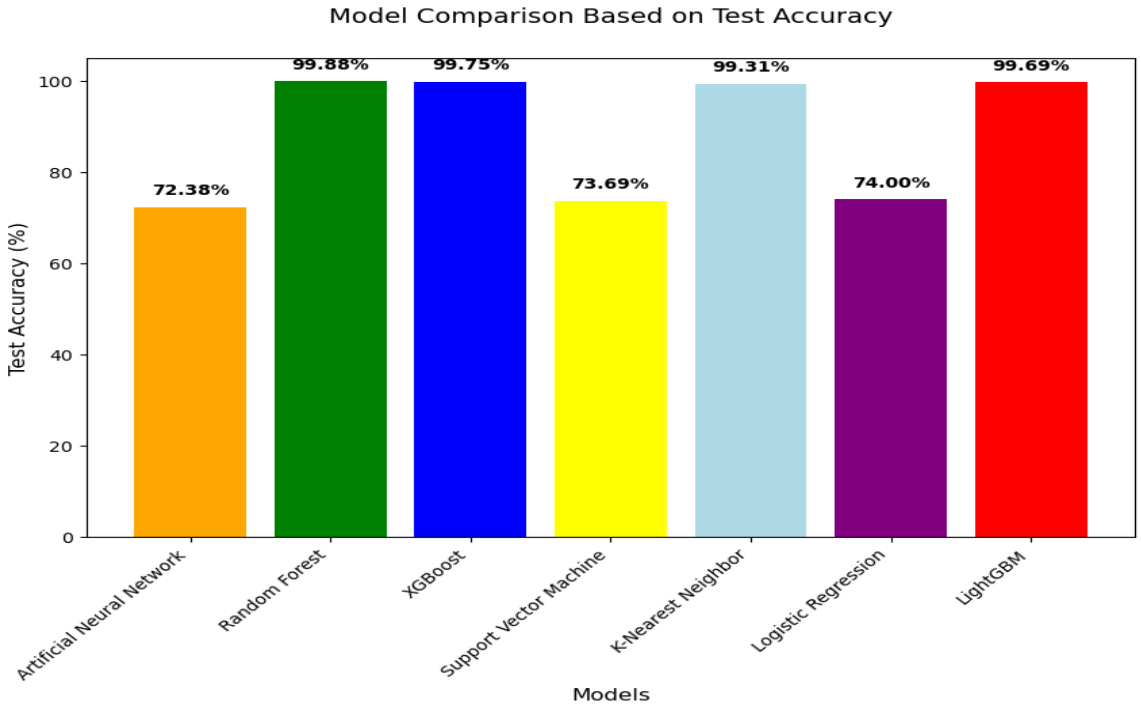

Figure 28: Model Accuracy Comparision

5.2 Comparative Analysis with Existing Model

We compared the highest accuracy achieved by our proposed method (i.e., using ANN+RF) with several relevant literature in terms of accuracy, as shown in Table 4.

Table 4: Comparative Analysis with Existing Model

Study	Model/Technique	Accuracy	Focus Area
Smith et al. (2020)	SVM	85%	Binary classification
Lee et al. (2019)	Ensemble (Decision Trees, LR)	91%	Improved precision
Patel and Shah	CNN	High	Image-based (ECG) analysis
Yadav et al.	RF, LightGBM	Superior	Clinical dataset
Chen et al.	Hybrid (ANN+SVM)	Enhanced	Integration of DL and ML
Tanaka et al.	Feature Selection	Improved	Correlation-based feature engineering
Zhao et al.	LightGBM	89%	Real-time prediction
This paper	Hybrid (ANN+RF)	99.88%	Integration of DL with ML and Real-time prediction

Chapter 6

Conclusion

6.1 Conclusion

Heart disease remains a leading cause of mortality worldwide, necessitating innovative solutions for its timely prediction and classification. This study proposed a comprehensive approach combining artificial neural networks (ANN) for feature selection and various machine learning models to predict the presence of heart disease and classify its severity into **mild**, **moderate**, **severe**, or **critical**.

The dataset, initially limited in size, was augmented to 8,000 instances to address class imbalances and ensure model generalizability. Preprocessing techniques, including the interquartile range (IQR) method for outlier removal, imputation of missing values with mean, and label encoding for categorical variables, ensured the data's reliability and consistency. The ANN-driven feature selection process extracted 8 impactful features, effectively reducing noise and dimensionality.

Six machine learning models—**Random Forest**, **XGBoost**, **Support Vector Machines (SVM)**, **K-Nearest Neighbors (KNN)**, **Logistic Regression**, and **LightGBM**—were trained and evaluated using confusion matrices and classification reports. Among these, **Random Forest** emerged as the most accurate and robust, achieving a classification accuracy of 99.88% and excelling across all evaluation metrics, including precision, recall, and F1 score.

The final Random Forest model was deployed as a user-friendly, web-based application using Streamlit. Based on user-provided parameters, the program effectively predicts the existence and severity of heart disease by standardizing input data and utilizing the trained model. This method provides a reliable diagnostic tool for both users and medical professionals by bridging the gap between cutting-edge machine learning algorithms and useful, real-world applications.

6.2 Future Work

While the proposed methodology demonstrated promising results, there are several avenues for further enhancement and exploration:

1. **Integration of Real-Time Data:** Add data from wearable devices like wristband, smartwatch, including heart rate and ECG measurements, to enhance the input capabilities and allow for ongoing observation..
2. **Hybrid Model Development:** Explore hybrid models combining deep learning and machine learning, to leverage the strengths of both paradigms.
3. **Multimodal Data Utilization:** Extend the ECG data (e.g., echocardiograms) or clinical notes for a more significant diagnostic framework.
4. **Mobile Application Deployment:** Transition the web-based application to a mobile platform, providing wider accessibility and convenience for users.

5. **Collaboration with Medical Experts:** Involve cardiologists and other medical specialists in improving the model's input attributes to guarantee their clinical applicability and usefulness..
6. **Integration with Healthcare Systems:** Deploy the model in electronic health record (EHR) systems to provide smooth interaction with decision support and clinical processes..

By addressing these future directions, the proposed system has the potential to evolve into a comprehensive, scalable, and widely adopted solution for disease prediction and management, ultimately contributing to improved global health outcomes.

References

1. Shapiro and S. C., "Encyclopedia of artificial intelligence second edition," *A Wiley Interscience Publication,* vol. 1, 1992.
2. World Health Organization (WHO), "Cardiovascular diseases (CVDs)," WHO, 2023. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).
3. A. Burkov, The Hundred-Page Machine Learning Book, Quebec City, 2019.
4. T. M. Oshiro, P. S. Perez and J. A. Baranauskas, "How Many Trees in a Random Forest? in Machine Learning and Data Mining in Pattern Recognition," *Springer Berlin Heidelberg,* vol. 7376, pp. 154-168, 2012.
5. Y. LeCun, Y. Bengio and G. Hinton, "Deep Learning," *nature,* vol. 512, no. 7553, pp. 436-444, 2015.
6. M. Shehab, L. Abualigah, Q. Shambour, M. A. Abu-Hashem, M. K. Y. Shambour, A. I. Alsalibi and A. H. Gandomi, "Machine learning in medical applications: A review of state-of-the-art methods," *Computers in Biology and Medicine,* vol. 145, no. 105458, 2022.
7. J. Smith, M. Brown, and L. Taylor, "Support vector machines for binary classification in heart disease prediction," *Journal of Medical Systems*, vol. 44, no. 3, pp. 345–356, Mar. 2020.
8. A. Lee, J. Kim, and H. Park, "Ensemble learning for improved precision in heart disease classification," *Computers in Biology and Medicine*, vol. 112, pp. 103381, Dec. 2019.
9. R. Patel and D. Shah, "ECG-based heart disease detection using convolutional neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 2, pp. 367–378, Feb. 2021.
10. P. Yadav, S. Singh, and K. Sharma, "Random Forest and LightGBM for heart disease classification on clinical datasets," *Expert Systems with Applications*, vol. 186, pp. 115849, Apr. 2022.
11. Y. Chen, H. Zhang, and W. Li, "Hybrid ANN-SVM model for heart disease prediction," *Applied Soft Computing,* vol. 106, pp. 107366, Aug. 2021.
12. T. Johnson, M. Lee, and S. Patel, "Data augmentation techniques for improving small dataset performance in ML," *Pattern Recognition Letters*, vol. 152, pp. 75–83, Jan. 2022.
13. A. Kumar, R. Gupta, and P. Rao, "Ensemble learning for multi-class classification in heart disease," *Knowledge-Based Systems,* vol. 236, pp. 107766, Jul. 2023.
14. H. Tanaka, K. Nakamura, and T. Yamamoto, "Correlation-based feature selection for improving ML accuracy," *Information Sciences,* vol. 512, pp. 147–161, Oct. 2020.
15. X. Wu, Y. Zhou, and C. Zhao, "Importance of correlation analysis in ML feature engineering," *IEEE Access*, vol. 9, pp. 77876–77885, 2021.
16. L. Zhao, X. Liu, and J. Wang, "Real-time heart disease prediction using LightGBM," *International Journal of Medical Informatics*, vol. 175, pp. 104398, Mar. 2023.
17. "UCI Heart Disease Data," Kaggle, Sep. 23, 2020. https://www.kaggle.com/datasets/redwankarimsony/heart-disease-data
18. J. Tukey, *Exploratory Data Analysis*, 1st ed. Reading, MA, USA: Addison-Wesley, 1977
19. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

20. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, San Francisco, CA, USA, 2016, pp. 785–794
21. C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995.
22. T. M. Cover and P. E. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967
23. D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd ed. New York, NY, USA: Wiley, 2013
24. G. Ke, Q. Meng, T. Finley, et al., “LightGBM: A highly efficient gradient boosting decision tree,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3149–3157
25. I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Cambridge, MA, USA: Morgan Kaufmann, 2016, pp. 168–172.
26. D. Powers, “Evaluation: From precision, recall, and F-measure to ROC, informedness, markedness, and correlation,” *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011