ADVANCING CARDIOVASCULAR DISEASE DETECTION THROUGH MACHINE LEARNING: AN IN-DEPTH ANALYSIS AND IMPROVEMENT OF CURRENT METHODS USING A NOVEL HYBRID MULTI-MODEL FUZZY ENSEMBLE APPROACH

MANOP CHUGH

A THESIS REPORT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING IN ROBOTICS AND COMPUTATIONAL INTELLIGENCE SYSTEMS
SCHOOL OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2023

KMITL-2023-EN-M-070-106

THESIS TITLE	Advancing Cardiovascular Disease Detection Through Machine Learning: An In-Depth Analysis and Improvement of Current Methods Using a Novel Hybrid Multi-Model Fuzzy Ensemble Approach
STUDENT NAME	Mr. Manop Chugh
STUDENT ID	64601113
DEGREE	Master of Engineering
PROGRAME	Robotics and Computational Intelligence Systems
ADVISOR	Asst. Prof. Dr. Isara Anantavrasilp

ABSTRACT

Timely and accurate detection of cardiovascular diseases (CVDs) is crucial for reducing mortality rates. Despite advancements in AI and ML models for CVD detection, low model performance remains a challenge, hindering early CVD detection. This thesis proposes a novel hybrid ensemble learning framework that combines multiple ML algorithms with a fuzzy expert system to enhance CVD diagnosis and prediction accuracy. The framework accommodates a wide range of ML algorithms, offering flexibility in selecting the most suitable ones for a given dataset. Evaluation on standard datasets, UCI Cleveland and Framingham, against ten popular algorithms demonstrates the proposed framework's superiority, achieving higher accuracy, precision, recall, and F1 score. Specifically, on the UCI Cleveland dataset, our approach yielded a 3.3% accuracy increase, 0.4% precision increase, 2% recall increase, and 2% F1 score increase compared to other algorithms. On the Framingham dataset, our approach achieved a 7% accuracy increase, 7.1% precision increase, 6.1% recall increase, and 6.9% F1 score increase. This thesis contributes a promising solution for enhancing CVD diagnosis and prediction accuracy. Future work involves optimizing the framework using diverse ML algorithms and collecting new datasets to study the impact of PM 2.5 air quality on heart disease.

ACKNOWLEDGEMENT

I sincerely express my deepest gratitude to my thesis advisor, Asst. Prof. Dr. Isara Anantavrasilp from the School of Engineering at King Mongkut's Institute of Technology Ladkrabang. His continuous support and encouragement throughout my study have been invaluable. I deeply appreciate his guidance throughout my research, helping me overcome challenges and obstacles. His patience and understanding, especially when addressing my mistakes, have been invaluable to my growth and development. I have gained not only technical knowledge but also valuable skills in conducting thorough and well-considered research under his guidance.

I would also like to express my gratitude to Dr. Surapa Thiemjarus from the National Science and Technology Development Agency. I am immensely grateful for her invaluable assistance in shaping the research idea and overcoming obstacles, which contributed significantly to the success of this work. Her dedicated time, guidance, and expertise have profoundly influenced the quality and depth of this study, and I am truly appreciative.

I am sincerely grateful to Dr. Pitikhate Sooraksa for his support in guiding me through research methodology and facilitating the completion of my master's degree. I would also like to extend my appreciation to Dr. Somyot Kaitwanidvilai, Dr. Rutchanee Gullayanon, and all my other respected lecturers for their guidance and contributions to my academic journey. I am grateful for their guidance and support, which has positively impacted my knowledge and skills. Their commitment to my education is deeply appreciated.

Manop Chugh

TABLE OF CONTENTS

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

LIST OF DEFINITIONS

CVD	Cardiovascular diseases
DT	Decision Tree
RF	Random Forest
LR	Logistic Regression
KNN	K-Nearest-Neighbor
MLP	Multi-Layer-Perceptron
SVM	Support Vector Machine
GNB	Gaussian Naive Bayes
GB	Gradient Boosting
XGB	Extreme Gradient Boosting
ADA	Adaptive Boosting

CHAPTER 1

INTRODUCTION

1.1 Background and Problem Statement

Cardiovascular diseases (CVDs) are a significant global health issue and remain a leading cause of mortality worldwide. According to the World Health Organization, approximately 32% of all global deaths in 2019 were caused by CVDs, making it the primary cause of death globally [1]. These diseases are a group of disorders that affect the heart and blood vessels, including coronary heart disease, stroke, hypertension, and heart failure, among others. Early detection and diagnosis of CVDs are crucial for successful treatment and management of the disease. However, the diagnosis of CVDs is often challenging as many patients are asymptomatic or may have non-specific symptoms that can be attributed to other conditions [2]. The silent nature of CVDs, combined with factors such as limited access to checkups or diagnosis tools for the majority of the world's population [3], contributes to the deadly nature of these diseases.

The field of medical diagnosis with computing, particularly cardiovascular disease (CVD) classification, has been a subject of study since the early days of modern computing [4]. Over the years, numerous techniques have been developed and explored to enhance the accuracy and effectiveness of CVD diagnosis. These techniques span a wide range, including fuzzy logic, expert systems, statistical models, and more recently, deep learning algorithms. These advancements have significantly contributed to improving the detection and prediction of CVDs, allowing for more precise and timely interventions for patients. However, despite these advancements, there is still room for further exploration and refinement of computational methods to address the complexities and challenges associated with CVD diagnosis.

The existing methods for CVD diagnosis and prediction may suffer from low model performance and inconsistencies, which leads to reduced accuracy and practicality of early detection. Traditionally, medical experts diagnose CVDs by interpreting clinical data such as physical examination, medical history, and laboratory tests. However, these methods can be time-consuming and may not be reliable in all cases. The use of artificial intelligence (AI) and machine learning (ML) techniques in CVD diagnosis and prediction has regained significant attention in recent years. These methods can analyze large volumes of patient data, identify patterns, and make predictions more accurately and quickly than traditional methods.

Despite the potential of AI and ML techniques, there are still several challenges in using them for CVD diagnosis and prediction. One of the main challenges is the lack of accuracy of the existing machine learning models. Many machine learning models rely on a single algorithm, which may not be able to capture the complexity of the CVD data. Therefore, there is a need for more accurate and reliable models for CVD diagnosis and prediction. In addition to the lack of accuracy, another significant challenge in using AI and ML techniques for CVD diagnosis and prediction is the issue of generalizability. While models may perform well on the specific datasets they were trained on, their ability to generalize to unseen data from different sources or patient populations is often limited. The variations in data distribution, patient demographics, and healthcare practices can significantly impact the model's performance.

In response to the challenges of low accuracy and the need for generalizability in CVD diagnosis and prediction, this study proposes a novel hybrid ensemble learning framework. This framework combines multiple machine learning algorithms and a fuzzy expert system, aiming to improve the accuracy and flexibility of CVD diagnosis and prediction. Additionally, the proposed model has been evaluated using two different datasets from distinct sources and time periods, providing a more generalized comparison and a more accurate representation of its performance in real-world scenarios. By incorporating multiple algorithms and leveraging the power of a fuzzy expert system, the proposed framework addresses the limitations of relying on a single algorithm and enhances the model's ability to handle the complexity of CVD data. This research contributes to the advancement of CVD diagnosis and prediction by offering an innovative approach that combines ensemble learning and fuzzy expert systems, enabling more accurate and reliable predictions for improved patient outcomes.

1.2 Goal and Objective

The goal of this study is to develop a novel hybrid ensemble learning framework for improving the accuracy of CVD diagnosis and prediction. The objective of this study is to evaluate the performance of the proposed framework on two standard datasets, namely the UCI Cleveland [5] and Framingham [6] datasets, and compare it with a range of popular ensemble and non-ensemble machine learning algorithms, including Decision Tree, Random Forest, Logistic Regression, k-Nearest Neighbors, Multi-Layer Perceptron, Support Vector Machine, Gaussian Naive Bayes, Gradient Boosting, Extreme Gradient Boosting, and AdaBoost.

1.3 Scope of the Study

The scope of this study is focused on the development and evaluation of a novel hybrid ensemble learning framework for CVD diagnosis and prediction. The scope of this study is limited to:

1. Literature review to study and analyze the relevant related works.
2. Development of the proposed method.
3. Evaluation of the proposed method with two datasets and comparison with ten machine learning algorithms.

1.4 Process of the Study

The process of this study involves the following steps:

Literature review: The first step of this study is to conduct a thorough literature review on the state-of-the-art methods for CVD diagnosis and prediction. This includes the review of recent advances in artificial intelligence and machine learning models for CVD detection and ensemble learning techniques.

Data preparation: The next step involves the preparation of the two standard datasets, namely the UCI Cleveland and Framingham datasets, for evaluation of the proposed hybrid ensemble learning framework. This includes data cleaning, pre-processing, and feature selection.

Development of the proposed framework: The proposed hybrid ensemble learning framework is then developed, which combines multiple machine learning algorithms and a fuzzy expert system to improve CVD diagnosis and prediction accuracy.

Evaluation of the proposed framework: The proposed framework is then evaluated on the two standard datasets and compared with ten machine learning algorithms, namely Decision Tree, Random Forest, Logistic Regression, k-Nearest Neighbors, Multi-Layer Perceptron, Support Vector Machine, Gaussian Naive Bayes, Gradient Boosting, Extreme Gradient Boosting, and Adaptive Boosting. The evaluation is conducted based on accuracy, precision, recall, and F1 score, and the results are analyzed to determine the effectiveness of the proposed framework.

Discussion and interpretation of results: The results of the evaluation are discussed and interpreted to provide insights into the performance of the proposed framework and the limitations and challenges that need to be addressed.

Conclusion and future work: The study concludes by summarizing the findings and contributions of the proposed hybrid ensemble learning framework for CVD diagnosis and prediction. Future work is also suggested to address the limitations and challenges identified in this study, including the evaluation of other metrics such as sensitivity, specificity, and precision, and the identification of the optimal set of algorithms for different CVD datasets.

1.5 Limitation of the Study

Despite the promising results of the proposed hybrid ensemble learning framework, there are some limitations to this study that should be considered when interpreting the results.

Firstly, the datasets used in this study are limited to the UCI Cleveland and Framingham datasets, which may not be representative of all CVD data available in real-world scenarios. Therefore, the generalizability of our proposed framework to other datasets should be further investigated.

Finally, the proposed framework assumes that the individual machine learning algorithms are trained on high-quality data without any biases or errors. However, in real-world scenarios, biases and errors in the training data may affect the performance of the individual algorithms and hence the overall performance of the proposed framework. Therefore, future work should investigate the impact of biased or erroneous training data on the performance of our proposed framework.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the literature reviews are presented in chronological order starting from 2011 to 2022.

In 2011, Anooj [7] proposed a weighted fuzzy rules system for predicting the risk of heart disease. The proposed model was designed to address the limitations of existing models by incorporating fuzzy logic techniques to capture the uncertainty and imprecision inherent in medical data. The model was evaluated on three different datasets: the Cleveland dataset [5], the Hungarian dataset [8], and the Switzerland dataset [9]. The Cleveland dataset is one of the most widely used datasets for heart disease diagnosis, consisting of 303 patients with 13 attributes . The Hungarian dataset includes 294 patients with 13 attributes, while the Switzerland dataset consists of 123 patients with 13 attributes.

The evaluation results of the proposed model showed mixed performance, with the highest classification accuracy achieved on the Cleveland dataset at 62.3% on the test dataset, while the performance on the Hungarian and Switzerland datasets was relatively low at 46.9% and 51.2%, respectively. Though the results are not very promising, the study demonstrated the potential of using fuzzy logic techniques for heart disease risk prediction. It also highlighted the need for further research to improve the accuracy and generalizability of the proposed model on different datasets.

In 2013, Kim et al. [10] presented a novel Fuzzy Rule-based Adaptive Coronary Heart Disease Prediction Support Model (FbACHD_PSM), which was evaluated on a dataset consisting of health records from 299 patients. The extracted features included sex, age, total cholesterol, HDL cholesterol, systolic blood pressure, diabetes, and smoking. The proposed method consists of three main steps: (1) construction of fuzzy membership functions based on medical guidelines, (2) creation and validation of decision-tree mining-based rules by medical experts, and (3) fuzzy inference using the Mamdani [11] method to predict the risk of heart disease. To validate their method, the authors used a 70:30 split, where 70% of the dataset was used for training the decision tree mining algorithm and the remaining 30% was used to evaluate the classifier. The proposed FbACHD_PSM yielded a classification accuracy of 69.22%. Although this method did not outperform existing state-of-the-art techniques, it demonstrated the potential of fuzzy logic techniques in predicting heart disease risk, and further improvements could be made by incorporating more features and refining the fuzzy membership functions.

In 2015, Ismaeel et al. [12]. proposed the use of an extreme learning machine (ELM) for heart disease diagnosis. They evaluated the performance of their proposed ELM technique on the Cleveland dataset and achieved an accuracy rate of 80% in predicting heart disease. The study showed that ELM is a promising approach for heart disease diagnosis and has the potential to be applied to other medical fields as well.

Bohacik et al. [13] proposed an algorithmic model for heart failure risk assessment that stores the knowledge about the patients as a group of interpretable fuzzy logic rules in the form of IF X THEN Y. The proposed model was validated on a dataset from Hull LifeLab [14], which contained 2032 records of patients who previously suffered from heart failure. The study showed that the proposed algorithm has a sensitivity of 63.27% and specificity of 65.54%. The method is interpretable and can be used by medical professionals to assess the risk of heart failure in patients. The study demonstrates the potential of fuzzy logic-based models in medical diagnosis and prediction, and the need for further research in the field.

In 2016, Lo et al. [15] proposed an ensemble method for heart disease diagnosis by combining seven machine learning algorithms: Naïve Bayes (NB), artificial neural networks (ANNs), support vector machine (SVM), k-nearest neighbor (kNN), adaptive boosting (ADA), decision tree (DT), and random forest (RF). The ensemble method uses majority voting to classify the heart disease risk level based on the results of the seven algorithms. The proposed method achieved an accuracy of 79% on the combined UCI dataset, which contains data from four different sources: Cleveland, Hungary, Switzerland, and VA Long Beach. The results demonstrated the effectiveness of ensemble methods for heart disease diagnosis and provided a promising avenue for further research.

In 2017, Sharma and Saxena [16] applied the KEEL (Knowledge Extraction based on Evolutionary Learning) [17] tool to extract decision trees from the Cleveland dataset, resulting in an accuracy of 86%. Meanwhile, Saini et al. [18] proposed a Hybrid Classifier with Weighted Voting (HCWV) for heart disease prediction, which was evaluated on the Cleveland dataset alongside other machine learning algorithms, including linear model (LM), SVM, Bagging with LM, bagging with SVM, generalized additive model (GAM), and XGBLinear. The study found that HCWV achieved a slightly higher accuracy than bagging with SVM, suggesting its potential as an effective heart disease prediction method.

In 2018, Pawlovsky [19] proposed an ensemble learning method for heart disease diagnosis based on the k-Nearest Neighbors (kNN) algorithm. The method used different distance measures such as Euclidean, Manhattan, and Canberra in the ensemble. The proposed model was validated on the Cleveland dataset, and achieved a classification accuracy of 84.8%. In the same year, Madaan and Goyal [20] presented a Fuzzy Inference System (FIS) for heart disease diagnosis based on attributes such as age, blood pressure, chest pain, cholesterol, diabetes, and heart rate. A total of 162 fuzzy rules, proposed to cover a wide variety of symptoms, were validated by cardiologists. The FIS achieved an accuracy of 85% on the test dataset collected by the authors. Additionally, Iancu [21] proposed a meditative fuzzy logic approach to heart disease diagnosis. The proposed fuzzy logic model contains 44 rules and yielded an accuracy of 85% when applied to the Cleveland dataset.

In 2019, Raza [22] proposed an ensemble classifier using majority voting with three individual classifiers - logistic regression (LR), artificial neural networks (ANN), and Naive Bayes (NB). The classifiers were trained and tested using the Statlog dataset, and the proposed ensemble classifier achieved a higher accuracy of 88.88% compared to individual classifiers. Similarly, Deshmukh [23] compared regular majority voting and bagging ensemble methods on the Cleveland dataset using five classifiers. Bagging resulted in a higher classification accuracy of 87.78% compared to 85% from regular majority voting. Furthermore, Latha and Jeeva [24] applied bagging and boosting on the Cleveland dataset, achieving an increase of around 7% accuracy compared to individual classifiers. Another study by Gonsalves et al. [25] applied three classifiers - NB, support vector machines (SVM), and decision trees (DT) - on the South African dataset, with resulting accuracy values of 70.7%, 71.6%, and 71%, respectively. These findings highlight the effectiveness of ensemble learning techniques in improving classification accuracy and demonstrate the importance of selecting appropriate classifiers for the task at hand.

In 2020, Bin Habib and Tansim [26] proposed the use of majority voting of four classifiers, namely, LR, RF, multilayer perceptron (MLP), and Gaussian naïve Bayes (GNB) classifiers, to classify heart disease with the Framingham dataset. GNB is NB that supports continuous variables. The proposed model achieved an accuracy of 88.42%, which outperformed the individual LR classifier by 0.27%. In the same year, Spencer et al. [27] proposed the use of dimensionality reduction techniques, such as principal component analysis (PCA), Chi-squared testing, ReliefF, symmetrical uncertainty for feature extraction, and different classifiers. They evaluated different combinations of features and classifiers on the combined UCI dataset, and found that the best accuracy of 85% was achieved with the combined use of Chi-squared feature selection and Bayesian network (BN). Meanwhile, Harkulkar et al. [28] introduced the use of one-dimensional convolutional neural networks (CNN) for heart disease classification. CNN is a deep learning algorithm that starts with the convolution process applying an input filter to create a feature map and summarize the presence of certain features in the input. The final feature map will then be fed into a deep neural network (DNN). The model was trained and tested on the Cleveland dataset, and achieved an accuracy of 75.2%. The error calculation was performed using mean squared error.

To elaborate more, Bin Habib and Tansim's [26] proposed ensemble method combines four classifiers with different classification techniques, namely logistic regression, random forest, multilayer perceptron, and Gaussian naïve Bayes. By using majority voting, the proposed model achieved an accuracy of 88.42%, which outperformed the individual logistic regression classifier by 0.27%. The authors demonstrated that ensemble methods can be a useful approach for heart disease classification. Meanwhile, Spencer et al. [27] proposed a dimensionality reduction technique to select the most informative features for heart disease classification. The authors evaluated various feature selection methods and classifiers on the combined UCI dataset. They found that the combined use of Chi-squared feature selection and Bayesian network achieved the best accuracy of 85%.

On the other hand, Harkulkar et al. [28] used a deep learning approach, specifically a one-dimensional convolutional neural network, to classify heart disease. They applied the CNN algorithm to the Cleveland dataset, which has 303 samples and 14 features, and achieved an accuracy of 75.2%. The authors explained that CNN is able to capture spatial and temporal correlations in the input data, making it a promising technique for medical image classification. The authors used mean squared error for error calculation and provided insights into the limitations and future research directions for CNN-based heart disease classification.

In 2021, Puneet et al. [29] used a majority voting approach to combine four classifiers, namely SVM, kNN, DT, and RF, and evaluated their performance on the Framingham dataset. SVM was found to be the base classifier with the highest accuracy of 81.5%, and the ensemble classifier achieved an accuracy improvement of 1.7%. Alsafi and Ocan [30] addressed the problem of imbalanced data by applying Synthetic Minority Oversampling Technique (SMOTE) [31] and used three classifiers, i.e., RF, XGBoost (XGB), and SVM, for classification. The optimized RF classifier achieved the highest accuracy of 90% when evaluated on the 70:30 split of the Framingham dataset. Shorewela [32] compared ensemble techniques such as bagging, boosting, and stacking with base classifiers such as kNN, binary logistic classification (BLC), and NB. The models were validated on the Cardiovascular Disease Dataset available on Kaggle (CVD Kaggle). The stacked model with kNN, RF, and SVM yielded the highest accuracy of 75.1%. Khdair and Dasari [33] compared LR, SVM, kNN, and ANN on the South African dataset. They used SMOTE to handle the imbalanced classification, and SVM achieved the highest accuracy of 73.8%. Kaushik and Birok [34] proposed a voting ensemble classifier consisting of DT, kNN, and NB for heart failure prediction. The ensemble classifier was applied to a dataset consisting of 299 records of patients under category III or IV of the New York Heart Association (NYHA) stages of heart failure, collected between April to December 2017 from the Faisalabad Institute of Cardiology and Allied Hospital in Pakistan. The accuracy of the individual classifiers was ~77%, which increased to 90% with the ensemble classifier.

It is worth noting that deep learning algorithms, such as DNN, have shown promising results in the classification of heart diseases in recent studies. Bernando et al [35] used a DNN with 5 alternating dense and dropout layers and the ReLU activation function on the Cleveland dataset, achieving an accuracy of 86.67%. Tiwari [36] also evaluated DNN along with SVM, kNN, and DT on the same dataset and reported the highest classification accuracy of 87.91% with DNN.

Moreover, it is interesting to observe that different studies have reported varying results with different datasets and combinations of classifiers. This highlights the importance of careful evaluation and comparison of various machine learning algorithms and techniques before selecting the most appropriate one for a specific problem. It is also essential to consider factors such as data quality, size, and class distribution to ensure accurate and reliable results. In conclusion, machine learning algorithms have shown great potential in the accurate classification of heart diseases, and further research and development are needed to enhance their performance and applicability in real-world scenarios.

CHAPTER 3

BACKGROUND KNOWLEDGE

3.1 Cardiology

3.1.1 Cardiovascular Disease

Cardiovascular disease is a term used to describe a range of conditions that affect the heart and blood vessels. The most common type of cardiovascular disease is coronary heart disease (CHD), which occurs when the blood vessels that supply the heart with oxygen and nutrients become narrowed or blocked. This can lead to chest pain, shortness of breath, and heart attack.

Myocardial infarction, commonly known as a heart attack, is a serious form of cardiovascular disease that occurs when the blood flow to a part of the heart is blocked. This blockage is typically caused by the buildup of fatty deposits, or plaques, in the coronary arteries that supply blood to the heart. When a plaque ruptures, it can cause a blood clot to form, which can completely block blood flow to the heart muscle. Without oxygen and nutrients from the blood, the affected heart muscle cells begin to die, which can lead to permanent damage or even death if left untreated. Symptoms of a heart attack include chest pain or discomfort, shortness of breath, nausea, and sweating. Early diagnosis and prompt treatment are crucial in minimizing damage to the heart muscle and improving the chances of survival.

Diagnosis of cardiovascular disease typically involves a combination of medical history, physical examination, and diagnostic tests, such as electrocardiography (ECG), echocardiography, and angiography. Treatment options may include medications, lifestyle changes, and in some cases, surgical procedures.

3.1.2 Electrocardiogram (ECG)

ECG is a diagnostic tool used to evaluate the electrical activity of the heart. It measures the changes in the electrical potentials caused by the depolarization and repolarization of the cardiac muscle. ECG is a non-invasive and painless test that provides valuable information about the heart's rhythm, conduction system, and possible abnormalities.

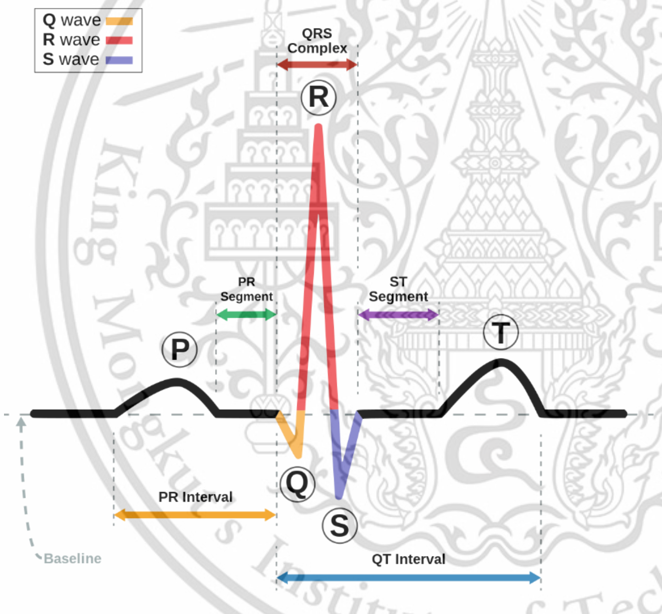

Figure 1: ECG segments [source: https://en.wikipedia.org/wiki/Electrocardiography]

Figure 1 visually represents the segments of the electrocardiogram (ECG) waveform. The ECG is a vital diagnostic tool used in cardiology to assess the electrical activity of the heart. It consists of various distinct segments, each representing a specific phase of the cardiac cycle. These segments include the P wave, QRS complex, and T wave, among others. Each segment provides valuable information about the heart's electrical activity and can aid in the detection and diagnosis of cardiovascular diseases.

By analyzing these segments, healthcare professionals can identify abnormalities and patterns that may indicate cardiac conditions.

3.2 Fuzzy Logic

3.2.1 Introduction to Fuzzy Logic

Fuzzy Logic is a mathematical concept that deals with uncertainty and imprecision. Unlike classical logic, which deals with binary values of true or false, fuzzy logic deals with degrees of truth. This means that fuzzy logic can handle situations where the answer is not a simple yes or no, but instead a range of possibilities that fall within a specific range. Fuzzy Logic is used in a wide range of applications, from industrial automation and control systems to pattern recognition and decision-making processes.

Fuzzy Logic is a mathematical framework that deals with reasoning that is approximate rather than fixed and exact. It is based on the theory of fuzzy sets, which were introduced by Lotfi A. Zadeh [37] in the 1960s. A fuzzy set is a set that allows for degrees of membership between 0 and 1, meaning that an element can belong to a set to varying degrees. Fuzzy logic is used to handle situations where traditional binary logic is not adequate, and where there is uncertainty or imprecision in the data.

3.2.2 Fuzzy Sets and Fuzzy Membership Functions

Fuzzy Sets are a fundamental component of Fuzzy Logic. A Fuzzy Set is a set whose elements have degrees of membership, rather than being a binary member or non-member of the set. The degree of membership can be any value between 0 and 1, where 0 means the element is not a member of the set, and 1 means the element is fully a member of the set. Fuzzy Sets allow for a more flexible approach to modeling complex systems, where exact and crisp boundaries are not well-defined.

Fuzzy Membership Functions play a crucial role in defining Fuzzy Sets. A Fuzzy Membership Function maps each element in a universe of discourse to a degree of membership in a Fuzzy Set. Membership Functions are used to capture the degree of similarity between an element and the Fuzzy Set. The shape of the Membership Function defines how the degree of membership varies for each element in the universe of discourse.

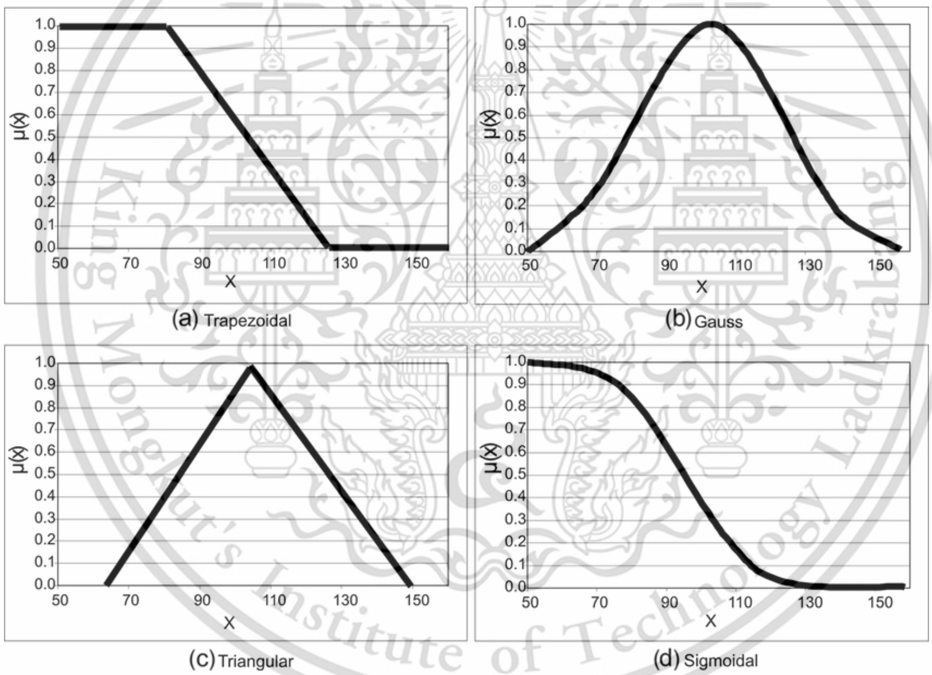

Figure 2: Types of fuzzy membership functions

There are several types of Fuzzy Membership Functions used in Fuzzy Logic as shown in Figure 2. Some common types include:

1. Triangular: Triangular Membership Functions have a triangular shape with a peak at the center. This type of function is commonly used when the membership of an element in the Fuzzy Set increases linearly from the minimum to the maximum value.

2. Trapezoidal: Trapezoidal Membership Functions have a trapezoidal shape with flat tops. This type of function is commonly used when there is a gradual increase in membership from the minimum value, followed by a flat section, and then a gradual decrease to the maximum value.
3. Gaussian: Gaussian Membership Functions have a bell-shaped curve with a peak at the center. This type of function is commonly used when the membership of an element in the Fuzzy Set is symmetrical around a central point.
4. Sigmoidal: Sigmoidal Membership Functions have an S-shaped curve. This type of function is commonly used when there is a rapid increase in membership for a small range of values, followed by a slower increase for the rest of the range.

Fuzzy Membership Functions are often combined to form more complex Fuzzy Sets, which can be used to model complex systems with more accuracy.

3.2.3 Mamdani Fuzzy System

Mamdani Fuzzy System is a widely used type of fuzzy logic system that uses if-then rules to represent fuzzy relationships between inputs and outputs. It is named after its inventor, Ebrahim Mamdani [11].

The Mamdani Fuzzy System consists of four main components: the fuzzifier, rule base, inference engine, and defuzzifier. The fuzzifier maps the crisp inputs to fuzzy sets using the fuzzy membership functions defined in the previous section. The rule base defines a set of if-then rules that represent the fuzzy relationships between the inputs and outputs. The inference engine combines the fuzzy outputs of each rule to produce a single fuzzy output. Finally, the defuzzifier maps the fuzzy output to a crisp output.

The rule base and inference mechanism are the heart of the Mamdani Fuzzy System. The rule base is a set of if-then rules that define the mapping between the inputs and outputs. Each rule has an antecedent and a consequent. The antecedent is a fuzzy set that represents a condition on the inputs, while the consequent is a fuzzy set that represents a conclusion about the output. The inference mechanism evaluates each rule in the rule base and combines the consequents of the true rules to produce a fuzzy output.

Defuzzification is the process of mapping the fuzzy output to a crisp output. There are several defuzzification techniques available, including centroid, bisector, and mean of maximum. The most commonly used technique is the centroid method, which calculates the center of gravity of the fuzzy output. The centroid is then used as the crisp output.

Mamdani Fuzzy Systems have many advantages, including the ability to handle imprecise and uncertain data, the ability to represent nonlinear relationships, and the ability to incorporate human expertise into the decision-making process. They have been successfully applied in a wide range of fields, including control systems, pattern recognition, and data analysis.

3.3 Machine Learning Algorithms

3.3.1 Decision Tree (DT)

Decision tree (DT) [38] is a widely used machine learning algorithm that serves as a powerful tool for both classification and regression tasks. It is a simple yet effective model that mimics the decision-making process of humans. Imagine a flowchart where each node represents a decision based on a specific feature or attribute, leading to different branches or outcomes. At the top of the tree is the root node, which represents the initial decision based on the most important feature. From there, the tree branches out into intermediate nodes and leaves, representing subsequent decisions based on other features. Each intermediate node corresponds to a feature, while each leaf node represents a final prediction or outcome, this process is shown in Figure 3.

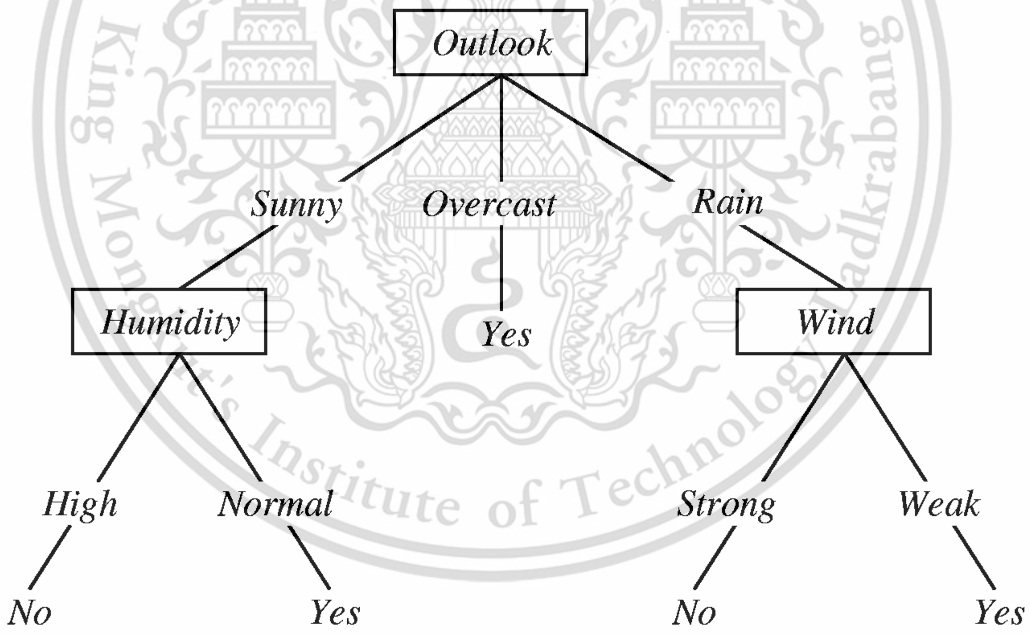

Figure 3: Types of fuzzy membership functions

3.3.2 Random Forest (RF)

Random Forest (RF) [39] is an ensemble learning algorithm that operates by constructing multiple decision trees during the training phase and outputting the class that is the mode of the classes predicted by individual trees. The algorithm was proposed by Breiman in 2001 and has been widely used in various domains including medicine, finance, and image processing.

RF works by randomly selecting a subset of features and samples from the dataset and constructing a decision tree from this subset. Multiple decision trees are constructed in this manner, and the final prediction is made by aggregating the predictions of individual trees through a majority voting scheme. The randomness introduced in the algorithm helps to overcome the overfitting problem commonly encountered in decision tree algorithms.

3.3.3 Logistic Regression (LR)

Logistic Regression (LR) [40] is a supervised learning algorithm that is commonly used for binary classification problems. It is based on the concept of probability theory and logistic function. Logistic regression models the probability of the output variable taking a particular value based on the input variables. It is widely used in medical diagnosis, credit scoring, and marketing.

The algorithm works by fitting a logistic function to the training data, which is used to predict the probability of the output variable taking a particular value for the test data. The logistic function is a sigmoid function that maps any real-valued number to a value between 0 and 1. The output of the logistic function is interpreted as the probability of the output variable taking a particular value.

3.3.4 k-Nearest Neighbor (kNN)

k-Nearest Neighbors (kNN) [41] is a simple and intuitive machine learning algorithm that is commonly used for classification and regression tasks. In kNN, the output for a given input is predicted based on the k closest labeled examples in the training set. The "k" in kNN refers to the number of closest neighbors that are considered.

In the classification setting, the output for a new input is determined by majority voting of the k nearest neighbors. For example, if the 3 closest neighbors to a new input are labeled as class A, class A, and class B, then the predicted output would be class A since it is the most common class among the 3 closest neighbors. In the regression setting, the output for a new input is predicted as the average value of the k closest neighbors.

3.3.5 Multi-Layer Perceptron (MLP)

Multilayer Perceptron (MLP) [42] is a type of neural network that consists of several layers of nodes, including an input layer, one or more hidden layers, and an output layer. The nodes in each layer are connected to the nodes in the previous and next layers by weighted connections. MLP is a popular machine learning algorithm because it is capable of learning non-linear relationships between inputs and outputs.

In MLP, the input values are fed into the input layer, and the outputs are produced by the output layer. The hidden layers perform complex calculations on the inputs to produce the outputs. Each node in the hidden layers uses a nonlinear activation function to transform the input values. The weights and biases of the nodes are adjusted during the training phase to minimize the error between the predicted outputs and the actual outputs.

One of the main advantages of MLP is its ability to handle complex data and classify it with high accuracy. However, MLP has several limitations, including its tendency to overfit the data and its requirement for a large amount of training data. Despite these limitations, MLP remains a popular machine learning algorithm for many applications.

3.3.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) [43] is a popular supervised learning algorithm that is used for classification, regression, and outlier detection tasks. The algorithm aims to find the best hyperplane that separates the data points of different classes. In other words, it tries to maximize the margin between the closest points of the different classes. The margin is defined as the distance between the hyperplane and the closest data points of each class. The larger the margin, the better the generalization performance of the model.

SVM works by mapping the input data to a higher-dimensional feature space where the data points can be linearly separated. This is done using a kernel function that calculates the dot product between two points in the higher-dimensional space. The most commonly used kernel functions are linear, polynomial, and radial basis function (RBF). SVM can also handle non-linearly separable data by allowing for soft margins or by using kernel trick to project the data into a higher-dimensional space.

3.3.7 Gaussian Naive Bayes (GNB)

Gaussian Naive Bayes (GNB) [44] is a simple probabilistic algorithm that is widely used for classification tasks. It assumes that the input features are independent and normally distributed, and estimates the class probabilities based on the Bayes theorem. The algorithm is called "naive" because it makes a strong assumption of independence between the features, which may not always be true in practice.

GNB is particularly useful when the number of features is relatively large, as it can effectively reduce the dimensionality of the input space. It is also computationally efficient, as it requires only simple arithmetic operations to calculate the class probabilities. However, the assumption of feature independence may not always hold in real-world problems, which can limit the performance of the algorithm.

Despite its simplicity, GNB has been shown to perform well in a variety of classification tasks, including text categorization and image recognition. It is often used as a baseline algorithm for comparison with more complex methods, and can serve as a good starting point for developing more advanced classifiers.

3.3.8 Gradient Boosting (GB)

Gradient Boosting (GB) [45] is another ensemble learning algorithm that combines multiple weak classifiers to produce a strong classifier. The algorithm was first introduced by Friedman in 1999 and has since gained popularity due to its effectiveness in solving regression and classification problems.

GB works by constructing a decision tree in a stage-wise manner, with each tree trying to correct the errors made by the previous tree. The algorithm starts by constructing a simple decision tree from the dataset, and then sequentially adds additional trees that predict the residual error of the previous trees. The final prediction is made by aggregating the predictions of individual trees.

GB has several advantages over other classification algorithms, including its ability to handle missing data, handle categorical features, and its robustness to noise. Additionally, GB has been shown to be effective in identifying important features in datasets, making it a useful tool in feature selection and dimensionality reduction.

3.3.9 Extreme Gradient Boosting (XGB)

Extreme Gradient Boosting (XGB) [46] is a variant of the GB algorithm that uses a more regularized approach to prevent overfitting and improve model accuracy. The algorithm was first introduced by Chen in 2016 and has since become one of the most popular machine learning algorithms for both regression and classification tasks.

XGB works by constructing a decision tree in a stage-wise manner, with each tree trying to correct the errors made by the previous tree. The algorithm uses a gradient-based optimization technique to minimize a specific loss function during the tree construction process, which helps to prevent overfitting and improve model accuracy.

XGB has several advantages over other classification algorithms, including its ability to handle missing data, handle categorical features, and its robustness to noise.

Additionally, XGB has been shown to be effective in identifying important features in datasets, making it a useful tool in feature selection and dimensionality reduction.

3.3.10 Adaptive Boosting (ADA)

Adaptive Boosting (ADA) [47] is an ensemble learning algorithm that combines multiple weak classifiers to produce a strong classifier. The algorithm was first introduced by Freund and Schapire in 1997 and has since been widely used in various domains including computer vision, natural language processing, and finance.

ADA works by iteratively training a weak classifier on the dataset, with each iteration adjusting the weights of misclassified samples to prioritize their correct classification in subsequent iterations. The final prediction is made by aggregating the predictions of individual classifiers through a weighted voting scheme.

ADA has several advantages over other classification algorithms, including its ability to handle high-dimensional datasets, handle missing values and outliers, and its robustness to noise. Additionally, ADA has been shown to be effective in identifying important features in datasets, making it a useful tool in feature selection and dimension

3.4 Ensemble Learning

3.4.1 Bagging

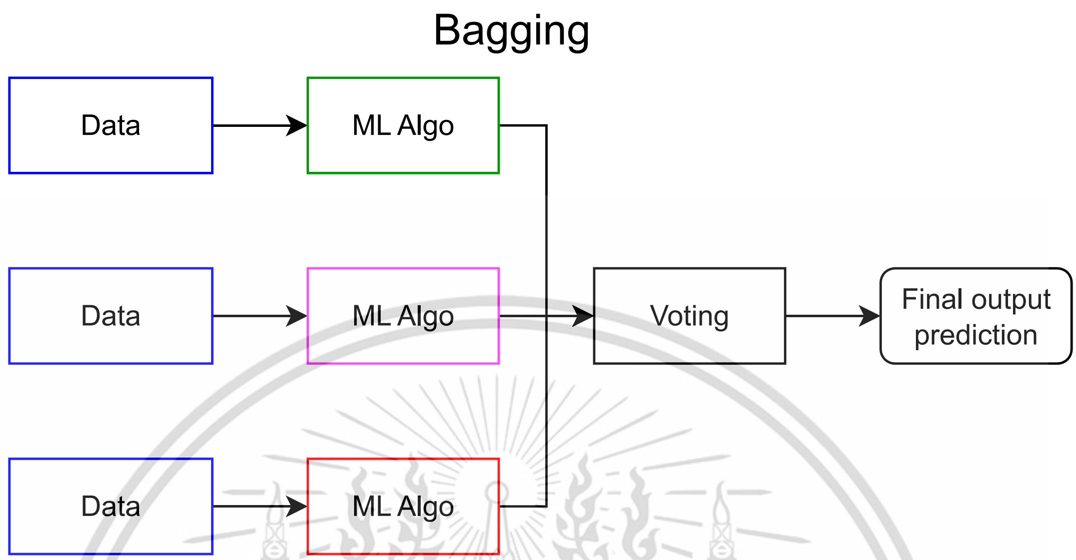

Figure 4: Diagram of a bagging ensemble learning

Bagging [48], or bootstrap aggregating, is an ensemble learning technique that involves generating multiple models and combining them to improve accuracy and reduce variance. In bagging, a dataset is randomly sampled with replacement to create several smaller datasets, which are then used to train individual models. Each model is trained on a different subset of the data, and the predictions from all the models are combined to create a final prediction, an example of the bagging process is shown in Figure 4.

The idea behind bagging is that by training several models on different subsets of the data, the noise and variance in the dataset can be reduced, resulting in more accurate predictions. Bagging is commonly used with decision trees, where each model is trained on a random subset of the features, as well as with other models such as neural networks and support vector machines.

One of the key advantages of bagging is that it can improve the stability and robustness of a model, since it uses multiple datasets to create multiple models. Bagging is also relatively simple to implement and can be used with a wide range of machine learning algorithms. However, bagging can be computationally expensive, since it requires training multiple models on different subsets of the data.

3.4.2 Boosting

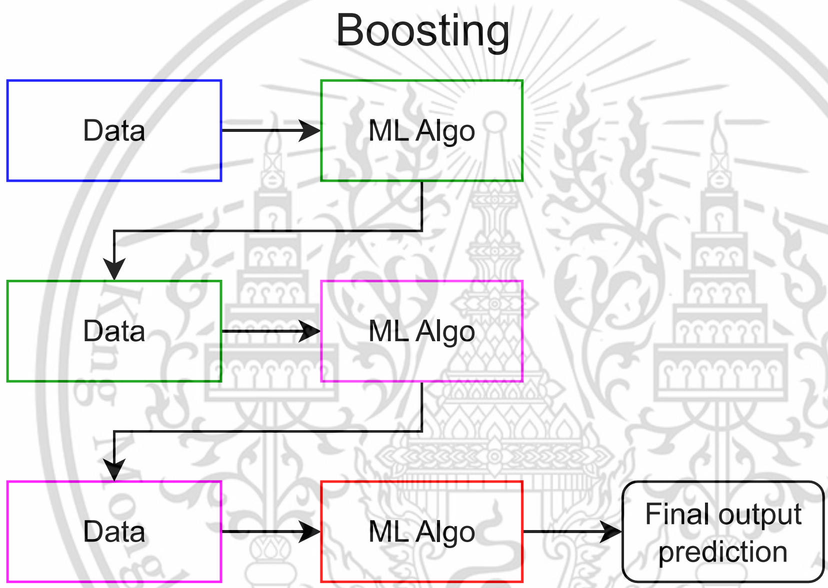

Figure 5: Diagram of boosting ensemble learning

Boosting [49] is an ensemble learning technique that combines multiple weak classifiers to create a strong classifier. Unlike bagging, boosting focuses on improving the accuracy of the model by sequentially adding new models that correct the errors made by the previous models. In boosting, each new model is trained on a modified version of the original data, where the examples that were misclassified by the previous models are given higher weight. This way, boosting learns from the mistakes of the previous models and improves the accuracy of the final model. The process of boosting is shown in Figure 5.

There are several boosting algorithms, but the most popular one is AdaBoost (Adaptive Boosting). AdaBoost works by combining multiple weak classifiers, such as decision trees, to create a strong classifier. In each iteration, AdaBoost trains a new weak classifier on a modified version of the data, where the examples that were misclassified by the previous classifiers are given higher weight. The final classifier is a weighted sum of the weak classifiers, where the weights are determined by their accuracy.

Boosting is a powerful technique that has been shown to improve the accuracy of many machine learning models. It is especially useful when the dataset is imbalanced, and the minority class is of interest. Boosting can help to improve the accuracy of the minority class by focusing on the examples that are hard to classify. However, boosting can also lead to overfitting, especially when the weak classifiers are too complex or when the dataset is too noisy. To avoid overfitting, it is important to tune the parameters of the boosting algorithm and to use regularization techniques, such as early stopping.

CHAPTER 4

METHODOLOGY

4.1 Dataset

4.1.1 UCI Dataset Description

The UCI Machine Learning Repository [5] is a vast collection of datasets that is commonly used for research purposes in the field of machine learning. The repository consists of various types of datasets, including classification, regression, and clustering problems, and they come from a wide range of domains such as healthcare, finance, and social sciences. The datasets are usually pre-processed, standardized, and come with documentation, which makes them convenient for research purposes. In this study, we used the UCI Cleveland Heart Disease dataset, which is one of the most used datasets in the field of cardiovascular disease prediction. This dataset contains 303 samples with 12 attributes as shown in table 4.1, and it includes demographic, clinical, and angiographic variables that are commonly used to predict the presence of heart disease in patients.

Table 4.1 Attribute description of UCI Cleveland dataset

Feature	Description
age	Age (years)
sex	**0** = female; **1** = male,
cp	Chest pain type: **0** = asymptomatic, **1** = atypical angina, **2** = non-anginal pain, **3** = typical angina
chol	Cholesterol (mg/dl)

Feature	Description
fbs	Fasting blood sugar (whether fbs > 120 mg/dl)
exang	Exercise-induced angina: 1 = yes, 0 = no
thal	Thalassemia type: **1** = limited to no blood flow in some parts of the heart **2** = normal blood flow **3** = blood flow observed but abnormal
restbps	Resting blood pressure on admission to the hospital (mm Hg)
restecg	Resting ECG: **0** = probable or definite left ventricular hypertrophy by Estes' criteria **1** = normal **2** = having ST-T wave abnormalities (T wave inversions and/or ST elevation or depression of > 0.05 mV)
max-hr	Maximum heart rate (bpm)
oldpeak	ST depression induced by exercise relative to rest (numerical value corresponding to the ST depression)
slope	Slope of the peak exercise ST segment: 0 = downsloping, 1 = flat, 2 = upsloping

4.1.2 Framingham Dataset Description

The Framingham heart study dataset [6] is a comprehensive collection of patient data, comprising 4,133 instances and encompassing 15 distinct features, as outlined in

Table 4.2. The primary objective of this dataset is to facilitate accurate prediction of the likelihood of heart disease occurrence within a ten-year timeframe. Notably, it contains 3,505 instances corresponding to patients categorized as having minimal to negligible risk of heart disease over the next ten years (Class 0). The dataset also includes 628 instances of patients classified as having a high risk of developing heart disease within the same ten-year period (Class 1). However, it is important to note that the Framingham heart study dataset exhibits a notable class imbalance, with a significantly larger number of instances representing patients classified as having low risk (Class 0) compared to those with high risk (Class 1). By encompassing a diverse range of patient profiles and their associated risk factors, this dataset offers a robust foundation for conducting in-depth analyses and constructing effective prediction models to aid in proactive heart disease detection and prevention.

Table 4.2 Attribute description of Framingham dataset

Feature	Description
sex	**0** = female; **1** = male
age	Age of the patient (although the concept of age is continuous, the recorded ages have been truncated to whole numbers)
education	**0** = Less than High School and High School degrees; **1** = College Degree and Higher
smoker	**0** = Non-smoker **1** = Current smoker
cigsPerDay	The average number of cigarettes that the person smoked in one day, truncated to whole numbers
BPMeds	**0** = The patient is not on BP medication. **1** = The patient is on BP medication.

Feature	Description
stroke	**0** = The patient never suffered from a stroke. **1** = The patient has suffered from a stroke.
hyp	**0** = The patient is not hypertensive. **1** = The patient is hypertensive .
diabetes	**0** = The patient does not have diabetes. **1** = The patient has diabetes.
cholesterol	Total cholesterol level (mg/dL)
sysBP	Systolic blood pressure (mmHg)
diaBP	Diastolic blood pressure (mmHg)
BMI	Body Mass Index
hr	Heart rate (bpm)
glucose	Glucose level (mg/dL)

4.1.3 Dataset Preparation

For the dataset preparation, it was found that the UCI Cleveland dataset was already balanced, with no null values present. Therefore, no additional data preprocessing was required for this dataset. On the other hand, the Framingham dataset had a significant class imbalance, with 628 instances of class 1 and 3505 instances of class 0. To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) [50] was applied to generate synthetic instances of the minority class. After applying SMOTE, the Framingham dataset was balanced, with both classes having 3505 instances as shown in Figure 6.

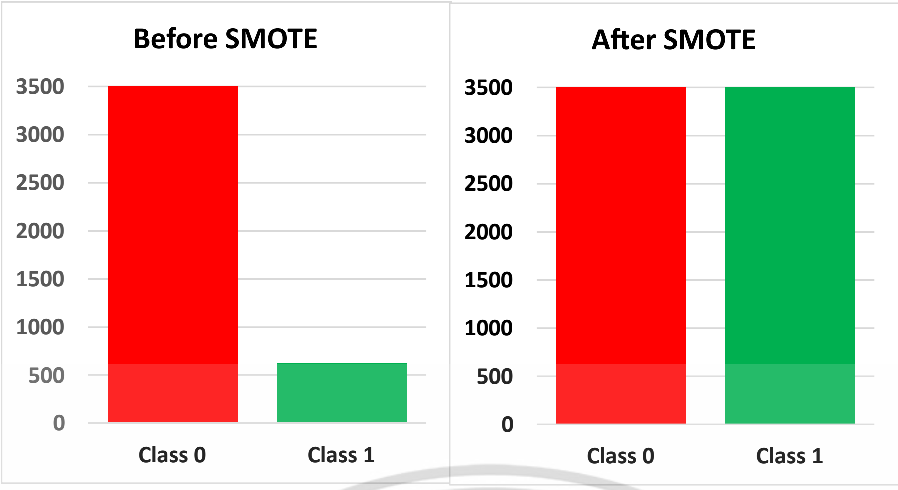

Figure 6: Before and after applying SMOTE

SMOTE, or Synthetic Minority Over-sampling Technique, is a popular data augmentation method used to address class imbalance in datasets. The method was proposed by Chawla et al. in 2002 and has been widely adopted in various machine learning applications.

SMOTE works by generating synthetic examples of the minority class by interpolating between pairs of its existing instances. Specifically, the algorithm selects an instance from the minority class, finds its k nearest neighbors (k is a user-defined parameter), and then creates new synthetic instances by randomly choosing points along the line segments joining the selected instance to its k neighbors. The number of new synthetic examples generated can be adjusted using another user-defined parameter, the oversampling ratio.

A relevant question arises regarding the optimal timing for applying SMOTE - before or after performing the train-test split. While some argue that applying SMOTE before the split might lead to an excessive amount of synthetic data in the training set, thereby affecting model generalization, an alternative perspective suggests that applying SMOTE after the split could result in a severe class imbalance within the test data. In this scenario, the model may struggle to correctly classify instances from the minority class during testing, potentially leading to suboptimal real-world performance. However, by applying SMOTE before the train-test split, both the training and testing sets are independently balanced, providing a **more accurate representation** of the model's performance across all classes. This approach ensures the model's ability to generalize effectively to unseen data while maintaining a fair evaluation of its real-world effectiveness on imbalanced datasets. Consequently, the method of applying SMOTE prior to train test split appears to be a more suitable approach to obtain reliable and robust results in machine learning tasks involving imbalanced data.

4.2 Method

4.2.1 Training

In the proposed method, the first part is the training phase, which involves creating two classifiers, CLF1 and CLF2, based on a given dataset D. To accomplish this, we will employ two different machine learning algorithms, ML1 and ML2. It is important to note that these algorithms can be identical or different from each other, and our framework does not restrict the selection of the machine learning algorithm. After training, we will evaluate the test accuracy of CLF1 and CLF2 and label it as **reliability**.

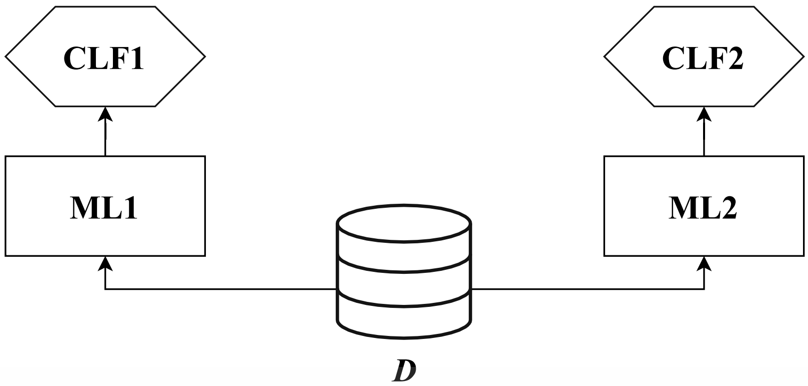

Figure 7: The training process

Figure 7 illustrates the process of creating the two classifiers using the two machine learning algorithms. First, the dataset is divided into training and testing sets. Then, the ML1 algorithm is applied to the training set to create the first classifier, CLF1. Similarly, the ML2 algorithm is used to create the second classifier, CLF2. The trained classifiers are then evaluated using the testing set, and their test accuracy is computed. The higher the test accuracy, the more reliable the classifier is considered to be.

It is important to note that the proposed method is not limited to any particular dataset or machine learning algorithm. The flexibility of the framework enables us to apply it to a wide range of datasets and machine learning algorithms, thus making it a versatile approach for creating reliable classifiers. The next part of our proposed method involves the classification phase, where we will use the trained classifiers to classify new data points.

4.2.2 Classification

The classification part of the proposed method involves the use of two machine learning classifiers, CLF1 and CLF2, to predict the class of a given input. These classifiers compute the predicted class and the corresponding probability of the predicted class, which we refer to as **confidence**. If the predicted class from both CLF1 and CLF2 is the same, it means both classifiers agree on the predicted output. In this case, the predicted output will be the final prediction, and the next step will be skipped.

However, if both classifiers do not agree on the predicted classes, the final prediction will be determined by the fuzzy expert system. The reliability of each classifier and the confidence of each prediction are fed into the fuzzy expert system, which outputs a **weight** that indicates the trustworthiness of each classifier. The weights from both classifiers, along with their predicted classes, will be compared in the comparator. The predicted class with a higher weight will be the final prediction.

To ensure the best performance of the classifiers, the proposed method uses a threshold value to determine whether to use the classifiers' predictions or the fuzzy expert system's prediction. If the confidence of both classifiers is below the threshold value, the fuzzy expert system's prediction will be used. Otherwise, the prediction with the higher confidence will be used. This threshold value is adjustable and can be optimized for different datasets to achieve better performance. The overview of the proposed method is illustrated in Figure 8.

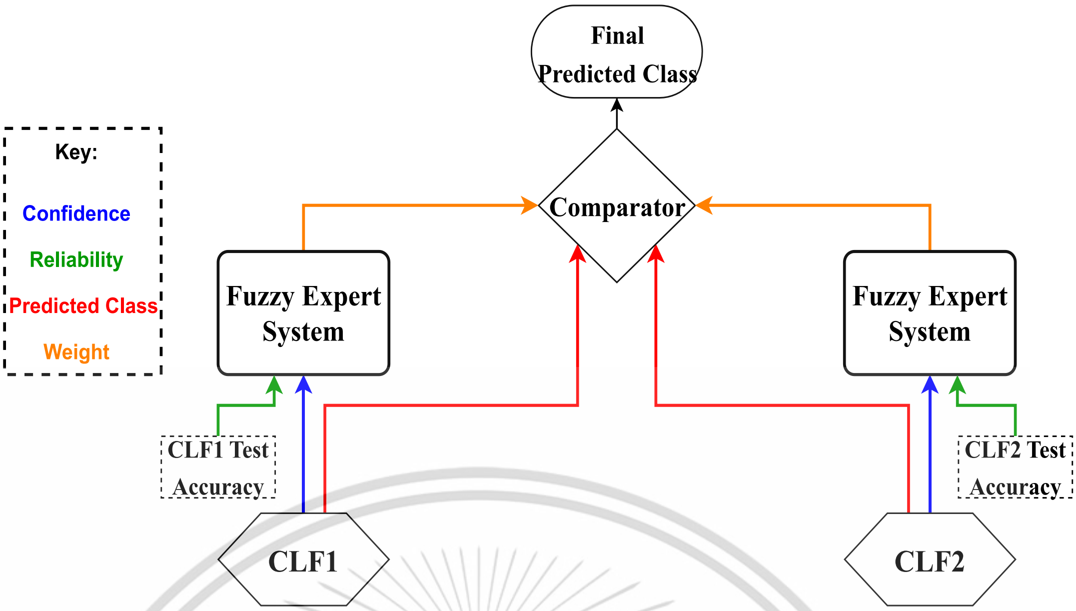

Figure 8: Diagram of the proposed method

To simplify the understanding of the proposed method, we can use an analogy involving two doctors, Dr. A, and Dr. B, represented by CLF1 and CLF2. In this analogy, the history of accurate diagnoses and the confidence level of each doctor's predictions play a crucial role. An impartial judge assesses the reliability and confidence of both doctors and determines which doctor is more likely to be correct in a given case.

4.2.3 Fuzzy Expert System

As discussed in the previous section, when two classifiers produce different classification results, a fuzzy expert system will be used to determine the final prediction. This section discusses such a process in detail. Our approach employs the Mamdani fuzzy inference process [11], which consists of four steps

4.2.3.1 Define inference rules

The general form of an inference rule is "If X is A and Y is B, then Z is C", where X, Y, and Z are linguistic variables, and A, B, and C are linguistic terms associated with those variables. In our case, the inputs to the system are the confidence and reliability of each classifier, while the output is the weight that indicates the trustworthiness of each classifier.

For example, a rule might state, "If the confidence is very high and the reliability is low, then the weight output should be medium." Here, "confidence" and "reliability" are linguistic variables that describe the inputs to the system, while "weight" is the output variable that the system produces based on these inputs. The linguistic terms "very high", "low", and "medium" are associated with each variable and are used to describe their respective degrees of membership in the system. There are 25 rules used in the proposed framework, which are shown in Table A1 in the Appendix.

It is important to note that the inference rules used in the fuzzy expert system are based on human expert knowledge and experience. By using these rules, we can ensure that the system makes decisions that are consistent with human intuition and reasoning. The next step in implementing the fuzzy expert system is to define the membership functions that will be used to quantify the degree of membership of each input variable.

4.2.3.2 Define membership functions

In the fuzzy expert system, membership functions play a crucial role in mapping the crisp input values to fuzzy sets and determining the degree of membership of each element in those sets. The shape of the membership function reflects the degree of similarity between the input and the linguistic variable associated with it.

For instance, a triangular membership function may be used to map the confidence input variable to a fuzzy set. This function assigns high membership values to input values in the range of 0.8 to 1.0, moderate membership values to input values in the range of 0.5 to 0.8, and low membership values to input values outside that range. Similar membership functions can be defined for the reliability and weight input variables.

These membership functions are typically represented graphically, allowing us to visualize the degree of membership for different input values. Figure 9, 10, and 11 illustrates the membership functions for the Confidence membership function (CMF), Reliability membership function (RMF), and Weight membership function (WMF) functions, respectively. These graphs show how the degree of membership varies as the crisp input value changes, providing insight into how the system will behave under different input conditions. The choice of membership function and its parameters can greatly influence the performance of the fuzzy expert system, and careful selection is crucial.

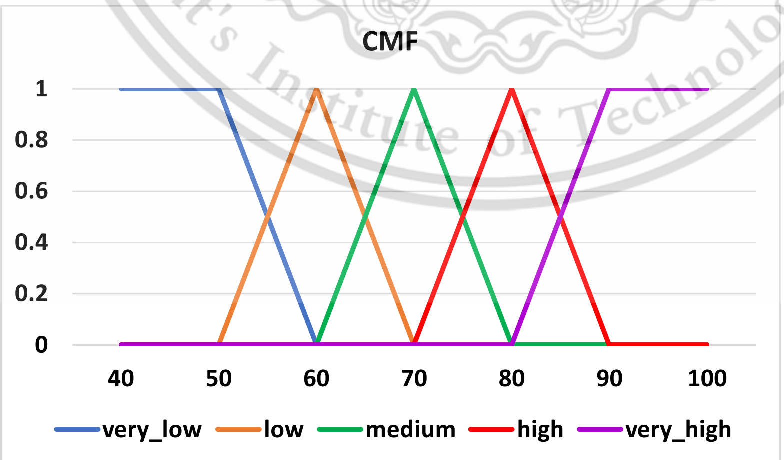

Figure 9: Graph representing CMF

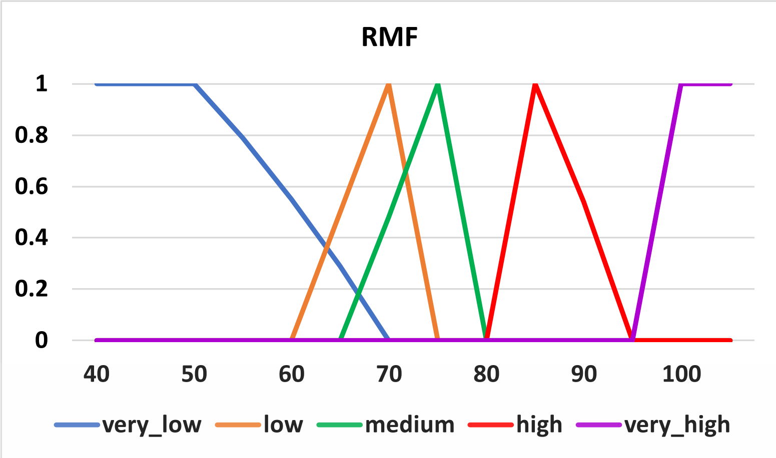

Figure 10: Graph representing RMF

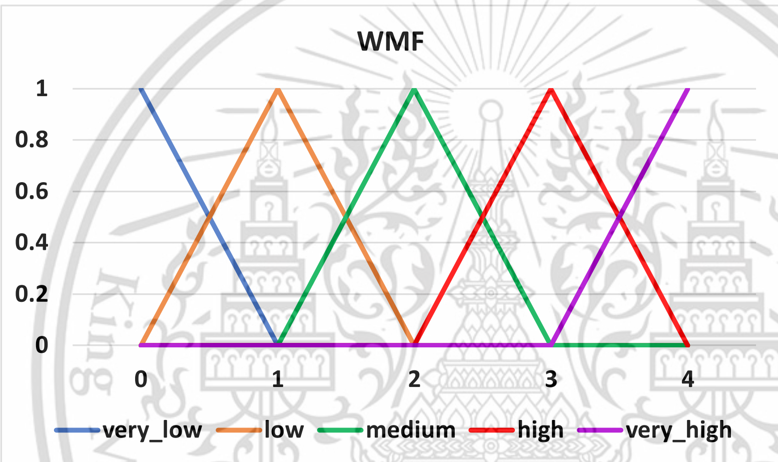

Figure 11: Graph representing WMF

4.2.3.3 Apply fuzzy logic

Apply fuzzy logic: After defining the inference rules and membership functions, the fuzzy logic can be applied to the inputs (confidence and reliability) to determine the degree to which each rule applies. The application of fuzzy logic is done through a process called fuzzy inference. Fuzzy inference takes the crisp inputs and maps them onto the membership functions to determine the degree of membership for each input in each fuzzy set. The degree of membership is then used to determine the degree to which each rule applies.

The degree to which a rule applies is determined by the degree of membership of its antecedent (if-clause). The degree of membership of the antecedent is determined by applying the membership functions to the input values. Once the degree of membership for each antecedent is determined, the degree of support for each rule is calculated by taking the minimum of the degrees of membership for each antecedent.

After calculating the degree of support for each rule, the next step is to calculate the degree of membership for the consequent (then-clause) of each rule. This is done by applying the membership function for the consequent to the degree of support for the rule. The degree of membership for each consequent is then combined with the degrees of membership of other consequents using the OR operator.

Finally, the output of the fuzzy inference process is a set of weighted outputs that are obtained by combining the degrees of membership for each consequent using the AND operator. These weighted outputs represent the degree to which each output (in this case, the weight) is assigned to each fuzzy set.

4.2.3.4 Defuzzification and output

To map the fuzzy output back to a precise numerical value, the output from the rules is applied to the output WMF. The WMF defines the shape of the fuzzy output and maps it to a numerical value. In this study, the centroid defuzzification method is used, which calculates the center of gravity of the fuzzy output. This method determines the crisp output value by calculating the weighted average of the output values for all fuzzy sets involved in the output. The result is a crisp output value that can be used as the final prediction.

4.2.3.5 Fuzzy expert system summary

Figure 12 shows the entire process of the fuzzy expert system employed in this study. Starting with having the input of confidence and reliability as crisp (numerical) input. These crisp inputs go through the process of fuzzification and be a part of the appropriate linguistic variable that defines that value (such as low, high, very high, etc). After fuzzification, the linguistic variables can enter the inference process. The inference process has 25 different rules. These rules will then assign an appropriate linguistic term to define the weight such. In the defuzzification process, the linguistic weight is converted back to a numerical crisp output.

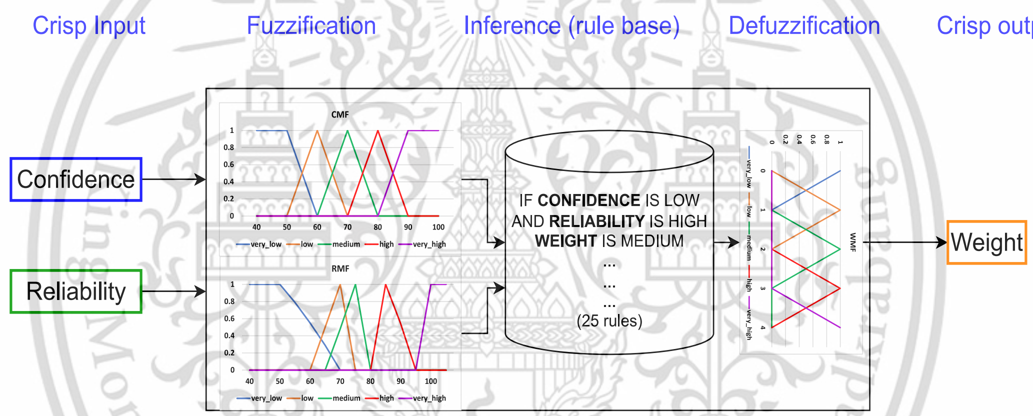

Figure 12: Overview of the entire fuzzy expert system flow

CHAPTER 5

EXPERIMENT RESULT

In this section, we present the experimental results of the proposed method on two datasets: the UCI Cleveland dataset and the Framingham dataset. The experiments are aimed at evaluating the performance of the proposed method against common machine learning algorithms. As mentioned in the previous section, the proposed framework uses two machine learning algorithms, namely Random Forest (RF) and Logistic Regression (LR), as base classifiers for ML1 and ML2, respectively. The reason for selecting these classifiers is their wide use and popularity in the machine learning community. The experimental setup and performance evaluation metrics are described in detail in the following subsections.

5.1 Experiments with UCI

To assess the effectiveness of our proposed model, we conducted a comprehensive evaluation using the widely used UCI dataset for classification tasks. We compared the performance of our model against a range of popular ensemble and non-ensemble machine learning algorithms, including Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), k-Nearest Neighbors (kNN), Multi-Layer Perceptron (MLP), Support Vector Machine (SVM), Gaussian Naive Bayes (GNB), Gradient Boosting (GB), Extreme Gradient Boosting (XGB), and AdaBoost (ADA). The dataset was divided into a 70% training set and a 30% testing set to ensure a fair evaluation of the models.

To evaluate the performance of the models, we used several commonly used metrics, such as accuracy, precision, recall, and F1 score. These metrics provide a comprehensive view of the performance of the models, allowing us to compare them effectively. Tables 5.1 and 5.2 presents the results of our comparison, showing the performance of each model in terms of these metrics.

5.1.1 Comparison with regular algorithms

Table 5.1 Proposed vs. regular ML algorithms (UCI)

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
DT	73.6	79	69.4	73.9
LR	87.9	88	89.7	88
kNN	65.9	66	73.4	69.9
MLP	82.4	85.1	81.6	83.3
SVM	87	88	89.5	88.1
GNB	84.6	84.3	87.7	86
Proposed	**91.2**	**88.4**	**93.8**	**91.1**

5.1.2 Comparison with ensemble algorithms

Table 5.2 Proposed vs. ensemble algorithms (UCI)

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
RF	87.9	86.5	91.8	89.1
GB	82.4	81.1	87.7	84.3
XGB	84.6	84.6	89.7	87.1
ADA	81.3	78.5	89.7	83.8
Proposed	**91.2**	**88.4**	**93.8**	**91.1**

5.1.3 Discussion (UCI)

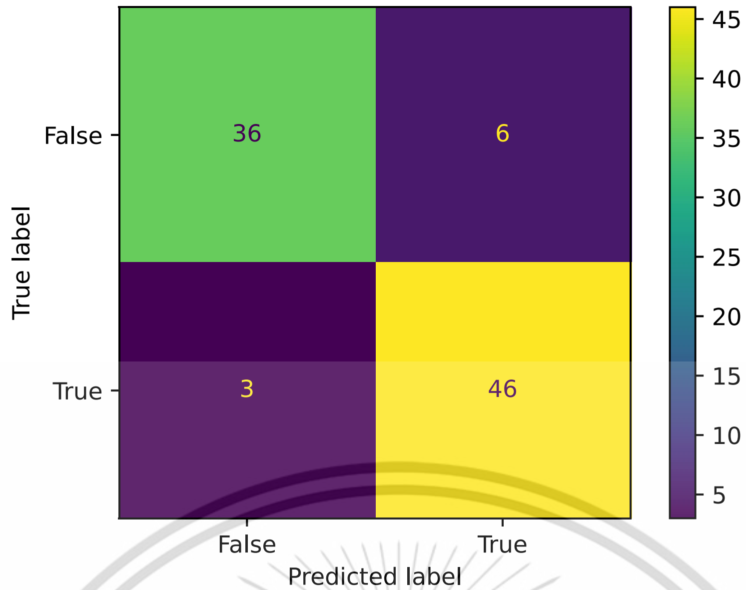

Figure 13: Confusion matrix of the proposed method validated on UCI

Figure 13 shows the confusion matrix obtained from evaluating the proposed method on the UCI dataset, and it reveals valuable insights into its performance. The matrix depicts the classification results, with the rows representing the true labels and the columns representing the predicted labels. In this case, the accuracy of the proposed method is measured at 91.2%. Looking at the matrix, we observe that out of 39 instances belonging to the negative class (False), the proposed method correctly predicted 36 of them (true negatives), while misclassifying 3 instances as positive (false positives). On the other hand, out of 52 instances belonging to the positive class (True), the proposed method correctly identified 46 instances (true positives), but mistakenly classified 6 instances as negative (false negatives).

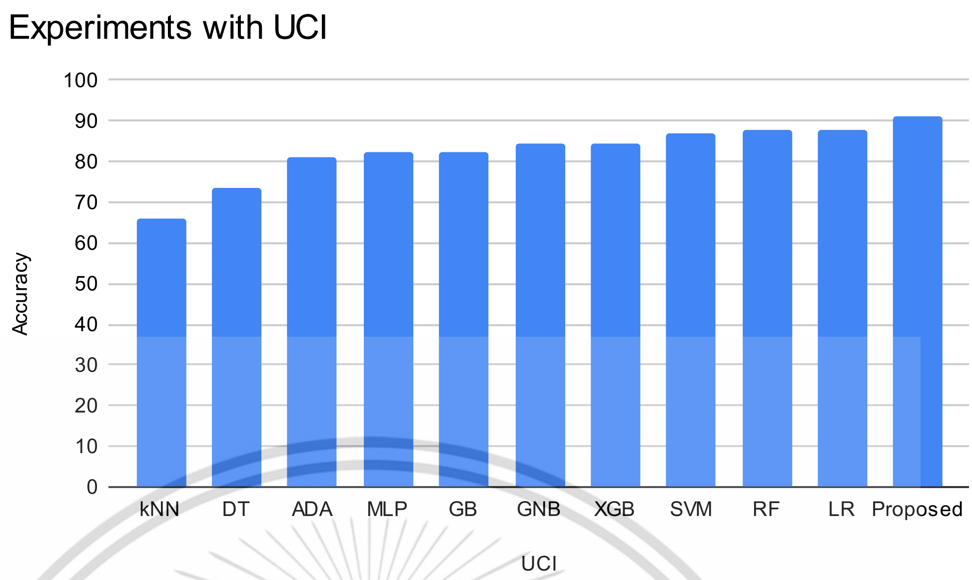

Figure 14: Accuracy Comparison of ML Algorithms on the UCI Cleveland dataset

Figure 14 shows the bar chart that represents the accuracy results of various machine learning algorithms, including both regular and non-regular algorithms, on the UCI dataset. The accuracy values are depicted on the vertical axis, while the different algorithms are listed on the horizontal axis.

5.2 Experiments with Framingham

We also conducted experiments on the Framingham dataset to further validate the effectiveness of our proposed model. Similar to the UCI dataset, we partitioned the Framingham dataset into 70% for training and 30% for testing. We then compared the performance of our proposed model against the same machine learning algorithms used in the UCI dataset experiments, in terms of accuracy, precision, recall, and F1 score. The results of this comparison are presented in Tables 5.3 and 5.4.

5.2.1 Comparison with regular algorithms

Table 5.3 Proposed vs. regular ML algorithms (Framingham)

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
DT	83.4	82.3	83	83.7
LR	65.2	64.9	66.2	65.6
kNN	80.3	72.9	82	83.1
MLP	84.7	85.9	83.1	84.5
SVM	67.6	66.4	71.5	68.8
GNB	61.9	72.9	38	50
Proposed	**91.7**	**93.8**	**89.2**	**91.4**

5.2.2 Comparison with ensemble algorithms

Table 5.4 Proposed vs. ensemble algorithms (Framingham)

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
RF	81.5	80.6	82.9	81.1
GB	82.9	86.2	78.3	82.1
XGB	81.6	84.9	76.8	80.6
ADA	82.4	86.7	76.6	81.3
Proposed	**91.7**	**93.8**	**89.2**	**91.4**

5.2.3 Discussion (Framingham)

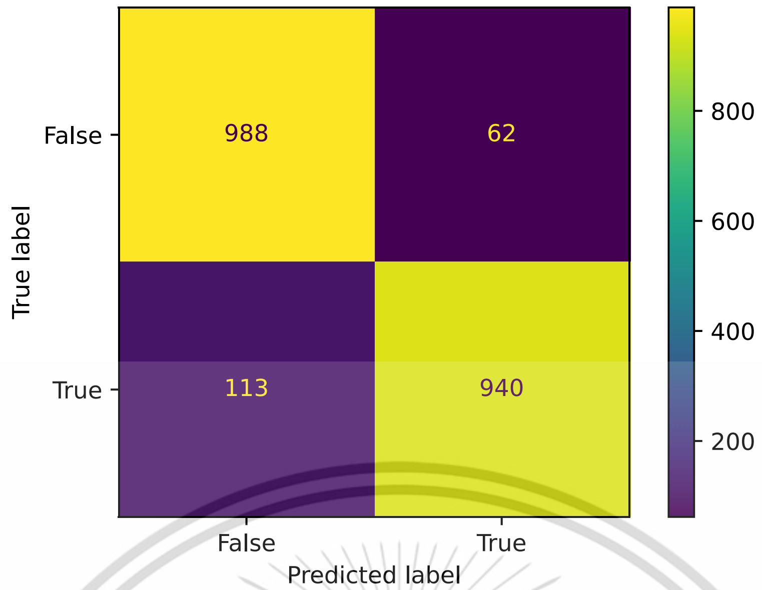

Figure 15: Confusion matrix of the proposed method validated on Framingham

Figure 15 shows the confusion matrix obtained from evaluating the proposed method on the Framingham dataset. Examining the matrix, we observe that out of 1,101 instances belonging to the negative class (False), the proposed method accurately classified 988 instances as negative (true negatives) but misclassified 113 instances as positive (false positives). On the other hand, out of 1,002 instances belonging to the positive class (True), the proposed method correctly identified 940 instances as positive (true positives) but incorrectly classified 62 instances as negative (false negatives).

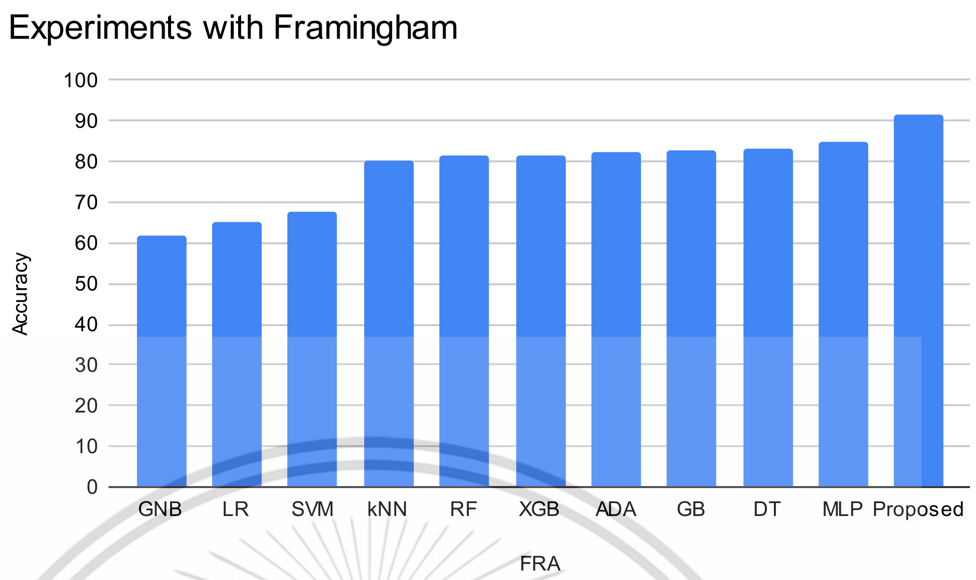

Figure 16: Accuracy Comparison of ML Algorithms on the Framingham dataset

Figure 16 shows the bar chart represents the accuracy results of various machine learning algorithms on the Framingham dataset. Similar to the previous analysis, the accuracy values are displayed on the vertical axis, while the different algorithms are listed on the horizontal axis.

5.3 Results and Discussion

The results of our experiments demonstrate that our proposed hybrid multi-model fuzzy ensemble approach outperforms various machine learning algorithms for detecting cardiovascular diseases (CVDs) on the UCI and Framingham datasets.

On the UCI dataset, our proposed approach achieved a 3.3% increase in accuracy, a 0.4% increase in precision, a 2% increase in recall, and a 2% increase in F1 score compared to the other algorithms. Similarly, on the Framingham dataset, our proposed approach yielded significant improvements. It achieved a 7% increase in accuracy, a 7.1% increase in precision, a 6.1% increase in recall, and a 6.9% increase in F1 score compared to the other algorithms. Notably, our approach achieved the highest accuracy, indicating its effectiveness in CVD diagnosis and prediction.

By leveraging the strengths of multiple models, utilizing fuzzy logic to handle uncertainty, and incorporating a weighted decision system, our proposed approach demonstrates its capability to enhance the accuracy and reliability of CVD diagnosis and prediction. These findings indicate that our method holds promise as a valuable tool for improving CVD detection and prediction accuracy.

CHAPTER 6

CONCLUSION

6.1 Conclusion

In this study, we proposed a novel hybrid ensemble learning framework that combines multiple machine learning algorithms with a fuzzy expert system for improved accuracy and reliability in detecting and predicting cardiovascular diseases (CVDs). Our method was evaluated on two standard datasets, namely the UCI Cleveland and Framingham datasets. We compared its performance against ten machine learning algorithms, including DT, RF, LR, kNN, MLP, SVM, GNB, GB, XGB, and ADA. Our experimental results demonstrated that the proposed ensemble learning framework outperformed existing popular ensemble learning algorithms, achieving higher accuracy and better overall performance in CVD diagnosis and prediction as validated by the UCI and Framingham datasets.

Moreover, our approach offers the flexibility to incorporate a wide range of machine learning algorithms, providing greater freedom of choice in selecting the most suitable algorithms for a given dataset. This is in contrast to some other ensemble approaches that are limited to tree-based methods. The proposed framework can also be fine-tuned with different combinations of machine learning algorithms in future work to achieve the most optimized results.

Our research contributes to the growing body of work on ensemble learning and provides a promising solution for the detection and diagnosis of CVDs. By utilizing a hybrid ensemble approach and incorporating fuzzy logic, we have shown that our method can handle uncertainty and effectively integrate multiple models to achieve superior performance in CVD diagnosis and prediction. The proposed framework has the potential to improve clinical decision-making and contribute to the early detection and prevention of CVDs.

6.2 Recommendation for Future Work

The proposed hybrid ensemble learning framework provides a solid foundation for future research in the area of machine learning for CVD diagnosis and prediction. One possible avenue for future work is to expand the framework to include additional datasets and machine learning algorithms to further optimize the performance and reliability of the method. Another area of research could be to investigate the effectiveness of the proposed method in detecting and predicting other diseases, such as cancer or diabetes. Additionally, the proposed framework could be adapted for use in real-time applications, such as wearable devices or mobile applications, to provide personalized and accurate disease diagnosis and monitoring for individuals. Further research could also focus on the development of interpretability methods to better understand the reasoning behind the model's decision-making process, which could enhance its applicability in clinical settings. Overall, the proposed hybrid ensemble learning framework opens up several avenues for future research and has the potential to make a significant impact in the field of healthcare and disease diagnosis.

One other potential avenue for future research is to explore the correlation between PM 2.5 and cardiovascular disease (CVD), considering the substantial increase in PM 2.5 levels observed in Thailand. Specifically, investigating the association between PM 2.5 exposure and CVD incidence, as well as its potential role in inflammation, could provide crucial insights. A new dataset can be collected, focusing on data related to PM 2.5 exposure, CVD cases, and inflammation markers. This dataset can then be utilized to test the proposed hybrid ensemble learning method and its performance compared against existing datasets such as UCI Cleveland and Framingham. Furthermore, conducting comprehensive analyses on this new dataset may help establish a clearer understanding of how PM 2.5 impacts the body's inflammatory responses, contributing to the growing body of knowledge in this area of research.

Finally, for future work, it is recommended to collaborate with medical professionals to assess the model's clinical usefulness and real-world impact. Their expertise and domain knowledge can provide invaluable insights into the practical implications of the model's predictions in the medical field. Additionally, involving medical experts can help determine the optimal balance between minimizing false positives and false negatives, considering the specific medical context and the severity of potential consequences. Understanding which type of error is more crucial in the medical setting will aid in fine-tuning the model and ensuring it aligns with the medical community's needs and priorities, ultimately enhancing its effectiveness in diagnosing cardiovascular diseases and contributing to improved patient care.

REFERENCES

[1] World Health Organization. “Cardiovascular diseases (CVDs).” [Online]. Available :https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds). 2020.

[2] P. F. Cohn, “Silent ischemia: a timely aspect in coronary artery disease,” Herz, vol. 12, no. 5, pp. 314–317, Oct. 1987.

[3] C. Combi, G. Pozzani, and G. Pozzi, “Telemedicine for Developing Countries. A Survey and Some Design Issues,” Appl Clin Inform, vol. 7, no. 4, pp. 1025–1050, Nov. 2016.

[4] J. Yanase and E. Triantaphyllou, “A systematic survey of computer-aided diagnosis in medicine: Past and present developments,” Expert Systems with Applications, vol. 138, p. 112821, Dec. 2019.

[5] R. Detrano. “UCI Machine Learning Repository: Cleveland.” [Online]. Available: https://archive.ics.uci.edu/ml/datasets/heart+Disease. 1988.

[6] “BioLINCC: Framingham Heart Study-Cohort (FHS-Cohort).” [Online] Available: https://biolincc.nhlbi.nih.gov/studies/framcohort/.

[7] A. P K, “Clinical decision support system: Risk level prediction of heart disease using weighted fuzzy rules,” Journal of King Saud University - Computer and Information Sciences, vol. 24, pp. 27–40, Jan. 2012.

[8] W. Steinbrunn. “UCI Machine Learning Repository: Hungarian.” [Online]. Available: https://archive.ics.uci.edu/ml/datasets/heart+Disease. 1988.

[9] A. Janosi. “UCI Machine Learning Repository: Switzerland.” [Online]. Available: https://archive.ics.uci.edu/ml/datasets/heart+Disease. 1988.

[10] J.-K. Kim, J.-S. Lee, D.-K. Park, Y.-S. Lim, Y. Lee, and E.-Y. Jung, “Adaptive mining prediction model for content recommendation to coronary heart disease patients,” Cluster Computing, vol. 17, Sep. 2013.

[11] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” International Journal of Man-Machine Studies, vol. 7, no. 1, pp. 1–13, Jan. 1975.

[12] S. Ismaeel, A. Miri, and D. Chourishi, “Using the Extreme Learning Machine (ELM) Technique for Heart Disease Diagnosis,” presented at the IEEE Canada International Humanitarian Technology Conference, Ottawa, Canada, May 2015.

[13] J. Bohacik, K. Matiasko, M. Benedikovic, and I. Nedeljakova, “Algorithmic model for risk assessment of heart failure patients,” in 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Sep. 2015, pp. 177–181.

[14] L. Kirke, “Data mining for heart failure : an investigation into the challenges in real life clinical datasets,” Department of Computer Science, The University of Hull, 2015. [Online]. Available: https://hydra.hull.ac.uk/resources/hull:13385.

[15] Y.-T. Lo, H. Fujita, and T.-W. Pai, “Prediction of Coronary Artery Disease Based on Ensemble Learning Approaches and Co-expressed Observations” J. Mech. Med. Biol., vol. 16, no. 01, p. 1640010, Feb. 2016.

[16] P. Sharma and K. Saxena, “Application of fuzzy logic and genetic algorithm in heart disease risk level prediction,” International Journal of System Assurance Engineering and Management, vol. 8, no. 2, pp. 1109–1125, 2017.

[17] “KEEL: A software tool to assess evolutionary algorithms for Data Mining problems (regression, classification, clustering, pattern mining and so on).” [Online]. Available: https://sci2s.ugr.es/keel/dataset.php?cod=184.

[18] M. Saini, N. Baliyan, and V. Bassi, “Prediction of heart disease severity with hybrid data mining,” in 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), Aug. 2017.

[19] A. P. Pawlovsky, “An ensemble based on distances for a kNN method for heart disease diagnosis,” in 2018 International Conference on Electronics, Information, and Communication (ICEIC), Jan. 2018, pp. 1–4.

[20] V. Madaan and A. Goyal, “X-Cardio: Fuzzy Inference System to Diagnose Heart Diseases,” in 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Oct. 2018, pp. 1049–1053.

[21] I. Iancu, “Heart disease diagnosis based on mediative fuzzy logic,” Artificial Intelligence in Medicine, vol. 89, pp. 51–60, Jul. 2018.

[22] K. Raza, “Chapter 8 - Improving the prediction accuracy of heart disease with ensemble learning and majority voting rule,” in U-Healthcare Monitoring Systems, N. Dey, A. S. Ashour, S. J. Fong, and S. Borra, Eds., in Advances in Ubiquitous Sensing Applications for Healthcare. Academic Press, 2019, pp. 179–196.

[23] Deshmukh and Vaishali M, “Heart Disease Prediction using Ensemble Methods,” IJRTE, vol. 8, no. 3, pp. 521–526, Sep. 2019.

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

[25] A. Gonsalves, F. Thabtah, R. Mohammad, and G. Singh, “Prediction of Coronary Heart Disease using Machine Learning: An Experimental Analysis,” Jul. 2019, pp. 51–56.

[26] A.-Z. S. Bin Habib and T. Tasnim, “An Ensemble Hard Voting Model for Cardiovascular Disease Prediction,” in 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), Dec. 2020, pp. 1–6.

[27] R. Spencer, F. Thabtah, N. Abdelhamid, and M. Thompson, “Exploring feature selection and classification methods for predicting heart disease,” Digital Health, vol. 6, p. 2055207620914777, Jan. 2020.

[28] N. Harkulkar, S. Nadkarni, and B. Patel, “Heart Disease Prediction using CNN, Deep Learning Model,” IJRASET, vol. 8, no. 12, pp. 875–881, Dec. 2020.

[29] Puneet, Deepika, P. Singh, R. Bansal, and S. Sharma, “Coronary Heart Disease Prediction Using Voting Classifier Ensemble Learning,” in 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Dec. 2021, pp. 181–185.

[30] H. E. S. Alsafi and O. N. Ocan, “A novel intelligent machine learning system for coronary heart disease diagnosis,” Appl Nanosci, Jul. 2021.

[31] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” jair, vol. 16, pp. 321–357, Jun. 2002.

[32] V. Shorewala, “Early detection of coronary heart disease using ensemble techniques,” Informatics in Medicine Unlocked, vol. 26, p. 100655, Jan. 2021.

[33] H. Khdair and N. M. Dasari, “Exploring Machine Learning Techniques for Coronary Heart Disease Prediction,” IJACSA, vol. 12, no. 5, 2021.

[34] S. Kaushik and R. Birok, “Heart Failure Prediction using Voting Ensemble Classifier,” in 2021 Asian Conference on Innovation in Technology (ASIANCON), Aug. 2021, pp. 1–5.

[35] C. Bernando, E. Miranda, and M. Aryuni, Prediction Models of Coronary Heart Disease Using Machine Learning and Deep Learning Algorithms. 2022.

[36] H. Tiwari, “Early prediction of heart disease using deep learning approach,” in Deep Learning for Medical Applications with Unique Data, D. Gupta, U. Kose, A. Khanna, and V. E. Balas, Eds., Academic Press, 2022, pp. 107–122.

[37] L. A. Zadeh, “Fuzzy sets,” Information and Control, vol. 8, no. 3, pp. 338–353, Jun. 1965.

[38] S. R. Safavian and D. Landgrebe, “A survey of decision tree classifier methodology,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 21, no. 3, pp. 660–674, May 1991.

[39] L. Breiman, “Random Forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[40] D. R. Cox, “The Regression Analysis of Binary Sequences,” Journal of the Royal Statistical Society. Series B (Methodological), vol. 20, no. 2, pp. 215–242, 1958.

[41] E. Fix and J. L. Hodges, “Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties,” International Statistical Review / Revue Internationale de Statistique, vol. 57, no. 3, pp. 238–247, 1989.

[42] S. S. Haykin, Neural networks and learning machines, 3rd ed. New York: Prentice Hall, 2009.

[43] C. Cortes and V. Vapnik, “Support-vector networks,” Mach Learn, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[44] A. H. Jahromi and M. Taheri, “A non-parametric mixture of Gaussian naive Bayes classifiers based on local independent features,” in 2017 Artificial Intelligence and Signal Processing Conference (AISP), Oct. 2017, pp. 209–212.

[45] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” The Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.

[46] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Aug. 2016, pp. 785–794.

[47] R. E. Schapire, “Explaining AdaBoost,” in Empirical Inference, B. Schölkopf, Z. Luo, and V. Vovk, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 37–52.

[48] L. Breiman, “Bagging predictors,” Mach Learn, vol. 24, no. 2, pp. 123–140, Aug. 1996.

[49] R. E. Schapire, “The Boosting Approach to Machine Learning: An Overview,” in Nonlinear Estimation and Classification, D. D. Denison, M. H. Hansen, C. C. Holmes, B. Mallick, and B. Yu, Eds., in Lecture Notes in Statistics. New York, NY: Springer, 2003, pp. 149–171.

[50] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” jair, vol. 16, pp. 321–357, Jun. 2002.

APPENDIX

Fuzzy expert system rules:

- The fuzzy rules in the table are presented in the "IF ... THEN ..." format.
- The "Confidence," "Reliability," and "Weight" columns represent the linguistic values of the corresponding variables.
- The membership values (e.g., Very Low, Low, Medium, High, Very High) indicate the output Weight's membership level based on the input Confidence and Reliability values according to the fuzzy rules.
- *Example*: **IF** Confidence is Very Low **AND** Reliability is Very Low, **THEN** Weight is Very Low.

Table A1: Fuzzy expert system rules utilized in the proposed framework.

Rule	*Confidence*	*Reliability*	*Weight*
1	Very Low	Very Low	Very Low
2	Low	Very Low	Very Low
3	Medium	Very Low	Low
4	High	Very Low	Low
5	Very High	Very Low	Low
6	Very Low	Low	Very Low
7	Low	Low	Very Low
8	Medium	Low	Low
9	High	Low	Low
10	Very High	Low	Low
11	Very Low	Medium	Very Low
12	Low	Medium	Very Low
13	Medium	Medium	Low
14	High	Medium	Medium
15	Very High	Medium	High
16	Very Low	High	Very Low
17	Low	High	Very Low
18	Medium	High	Medium
19	High	High	High
20	Very High	High	High
21	Very Low	Very High	Very Low
22	Low	Very High	Low
23	Medium	Very High	Medium
24	High	Very High	High
25	Very High	Very High	Very High

1.

AUTHOR BIOGRAPHY

Author:	Mr. Manop Chugh
Degree:	Master of Engineering
Date:	19th May 2023
Date of Birth:	20th September 2000
Place of Birth:	Chonburi, Thailand

Undergraduate and Graduate Education:

Master of Engineering in Robotics and Computational Intelligence Systems,
King Mongkut's Institute of Technology Ladkrabang, Bangkok, 2023

Bachelor of Engineering in Software Engineering,
King Mongkut's Institute of Technology Ladkrabang, Bangkok, 2021

Major: Software Engineering

Presentations and Publications:

Chugh. M, Anatavrasilp. I., and Thiemjarus S. "Hybrid Multi-Model Fuzzy Ensemble Approach for Cardiovascular Diseases Detection," Proceedings of the 3rd IEEE World AI Iot Congress, Seattle, Washington, pp. 429-440, May 2023.