Republic of Iraq

Ministry of Higher Education and Science Research

University of Babylon

College of Science for Women

Department of Computer Science

Secure Cardiac Diagnostic System Based on Machine Learning

A Thesis

Submitted to the Council of College of Sciences for Woman,

University of Babylon in Partial Fulfillment of the Requirement

For Degree of Master of Science in Computer Science

By

Randa Shaker Abd_alhussain

Supervised by

Prof. Dr. Hadab Khalid Obayes

Dr. Farah Mohammed Hassan Al-Shareefi

2023 A.D. **1445 A.H.**

بسم الله الرحمن الرحيم

(قالوا سبحانك لا علم لنا إلا ما علمتنا انك أنت العليم الحكيم)

صدق الله العلي العظيم

سورة البقرة – الآية(32)

Dedication

To

My Twelfth Imam Al-Hujjah Ibn Al-Hasan, who will fill the earth with equity and justice after it has been filled with injustice and oppression,

To

those who sacrificed themselves in defense of their land

Our righteous martyrs,

To

my family,

I dedicate this humble effort

Randa Shaker ... (2023)

Supervisors Certification

we certify that this thesis entitled "***Secure Cardiac Diagnostic System Based on Machine Learning***" was done by (*Randa Shaker Abd_Alhussain*) under our supervision.

Signature:

Name: ***Prof. Dr. Hadab Khalid Obayes***

Date: / / *2023*

Address: *College of Education for Humanities Studies, University of* Babylon

Signature:

Name: ***Dr. Farah Mohammed Hassan Al-Shareefi***

Date: / / *2023*

Address: *College of Science for Women, University of Babylon*

The Head of the Department Certification

*In view of the available recommendations, I forward the dissertation entitled "**Secure Cardiac Diagnostic System Based on Machine Learning**" for examining committee.*

Signature:

Name: ***Dr. Saif M. Kh. Al-Alak***

Date: / / 2023

Address: College of Science for Women, University of Babylon

Acknowledgments

All thanks and praise to Allah, the Lord of the world, who gave me courage and enabled me to achieve this work.

My thanks and gratitude go to my supervisors ***Dr. Hadab Khalid Obayes*** and ***Dr. Farah Mohammed Hassan Al-Shareefi*** for the support and guidance they have given me and the effort and time to complete this research.

Thanks, and gratitude to all my professors and all the staff of the Department of Computer Sciences\College of Sciences for Women\University of Babylon for their help.

Randa Shaker ... (2023)

Abstract

This thesis focuses on three issues concerning the diagnosis of heart disease and the security of diagnostic data. First, considering the key distribution problem associated with symmetric-key encryption schemes. Second, choosing an appropriate machine-learning technique capable of accurately and timely diagnosing heart problems. Finally, extending the training dataset to improve the deep learning technique's prediction performance.

As a result, This thesis builds a three-stage automated heart disease diagnostic system. The first stage addresses the symmetric key issue by utilizing the Three-Pass protocol for the distribution of a secret key. This protocol is chosen due to its ability to facilitate secure message transmission between parties without requiring prior knowledge or key sharing. This stage is evaluated by launching dictionary and Man-In-The-Middle (MITM) attacks. The dictionary attack fails because the keys are randomly produced using numbers, characters, and other symbols not in the English lexicon. MITM attack succeeds, however adding the verification step of not sending the same messages to honest participants can prevent it.

In the second stage, the K-Nearest Neighbor (KNN) and Random Forest (RF) algorithms are analyzed using various performance criteria to solve the second problem. A data set of vital markers is used to classify the patient as having heart illness at this stage. The accuracy scale findings show that the RF algorithm (99%) outperformed KNN (96%).

While, the insufficient training datasets are addressed in the third stage to increase Long Sort Term Memory (LSTM) performance. In essence, the Conditional Tabular Generative Adversarial Network approach is applied in order to expand the training data set. Data set rows grow from 299 to 5000. The experimental findings reveal that LSTM prediction accuracy increases from 65% to 99%.

Table of Contents

Chapter One **General Introduction**

1.1 Introduction 2
1.2 Problem Statement 4
1.3 Aims of Thesis 5
1.4 Contributions of Thesis 5
1.5 Related Work 5
1.6 Thesis Outline 12

Chapter Two **Theoretical Background**

2.1 Introduction 14
2.2 Cryptographic Primitives 14
2.2.1 Public-Key Cryptography 14
2.2.2 Symmetric-Key Cryptography 15
2.3 Advanced Encryption Standard 15
2.4 Cryptographic Protocol 16
2.4.1 Three –Pass Protocol 17
2.4.2 Protocol Attacks 18
1) Man-in-the-Middle Attack 18
2) Dictionary Attack 18
2.5 Cardiology Diagnostic Systems 19
2.6 Classification 19
2.6.1 Machine Learning Models 21
1) The K-Nearest Neighbor 21
2) The Random Forest 23
2.7 Deep Learning (Deep Learning and Deep Neural Network) 25
2.7.1 Recurrent Neural Network (RNN) 26
2.7.2 Long Short Term Memory (LSTM) 28
2.8 Data Augmentation 31

2.9 Conditional Tabular Generative Adversarial Network (CTGAN)........... 32
2.9.1 Metrics For CTGAN Performance Evaluation............................ 34
2.10 Performance Measures 35
Chapter Three *The Proposed System*
3.1 Introduction 39
3.2 Structure of the Proposed System 39
3.2.1 Security Provision Stage 43
3.2.2 Patient State Detection Stage 45
3.2.2.1 Pre-processing Stage……………………..………………47
3.2.2.2 Splitting Data Stage………………………………………47
3.2.2.3 Detection Stage…………………...…………………………48
3.2.3 Heart Attack Prediction Stage 48
3.2.3.1 Pre-Prpcessing Stage and Splitting Data Stage 49
3.2.3.2 Data Augmentation Stage 49
3.2.3.3 Prediction Stage 50
Chapter Four *Experimental Results and Evaluation*
4.1 Introduction 52
4.2 Hardware and Software Requirements 52
4.3 Results of the Propoed System... 52
4.3.1 Dataset Setting 52
4.3.2 Security Provision Stage 54
4.3.3 Patient State Detection Stage 57
4.3.4 Heart Attack Prediction Stage.. 67
Chapter Five *Conclusion and Suggestions for Future Works*
5.1 Conclusions 72
5.2 Suggestions for future works 73
References 74

List of Figures

Figure No.	Title of Figure	page
2.1	A General Process of AES algorithm	16
2.2	The Main Concept of Man-In-The-Middle Attack	18
2.3	The Main Concept of Dictionary Attack	19
2.4	The KNN Algorithm Structure	22
2.5	The RF Algorithm Structure	24
2.6	Deep RNN Structure	27
2.7	Long Short Term Memory Neural Network Algorithm	28
2.8	CTGAN Schema	33
2.9	Confusion Matrix	37
3.1	Schematic Flow Chart Diagram of the Proposed System	40
3.2	Schematic Flow Chart of the Security Provision Stage	41
3.3	Schematic Flow Chart of the Patient State Detection Stage	42
3.4	Schematic Flow Chart of the Heart Attack Prediction Stage	43
3.5	The Conceptual Scheme of Three Pass Protocol	45
4.1	The Accuracy of Random Forest	58
4.2	The Confusion Matrix of Random Forest where n_estimator = 1	59
4.3	The Confusion Matrix of Random Forest where n_estimator = 2	59
4.4	The Confusion Matrix of Random Forest where n_estimator = 3	60
4.5	The Confusion Matrix of Random Forest where n_estimator = 5	60
4.6	The accuracy of K_Nearest Neighbor	61
4.7	The Confusion Matrix of KNN where k = 3	62
4.8	The Confusion Matrix of KNN where k = 5	62
4.9	The Confusion Matrix of KNN where k = 7	63
4.10	The Confusion Matrix of KNN where k = 9	63
4.11	The Accuracy of Support Vector Machine	64
4.12	The Confusion Matrix of Support Vector Machine	65
4.13	The Accuracy of Naïve Bayes	66
4.14	The Confusion Matrix of Naïve Bayes	66
4.15	Performance Evaluation of RF, KNN, SVM and NB Algorithms	67
4.16	The Accuracy of LSTM Before Data Augmentation	68
4.17	The Accuracy of LSTM After Data Augmentation	70
4.18	The Confusion Matrix of LSTM	70
4.19	The Performance Comparison Between LSTM Before and After Data Augmentation	71
4.20	The Performance Comparison Proposed Work with Some Related Work	72

List of Table

Table No.	Title of Table	Page
1.1	Summary of Related Work	11
3.1	First Dataset Attributes Names	46
3.2	Second Dataset Attributes Names	48
4.1	The Employed Dataset in the Security Provision Stage	53
4.2	Avalanche Effect Test Results	56
4.3	The Effectiveness of the Random Forest Classification Method	57
4.4	The K-Nearest Neighbor Classification Technique's Performance Results	61
4.5	The Support Vector Machine Classification Technique's Performance Results	64
4.6	The Naïve Bayes Machine Classification Technique's Performance Results	65
4.7	The Performance Results of the Long Short-Term Memory Prediction Technique Before Data Augmentation	68
4.8	The Performance Results of the Long Short-Term Memory Prediction Technique After Data Augmentation	69
4.9	Comparison Table with Some Related Work	71

List of Abbreviations

Abbreviations	Meaning
AI	Artificial Intelligence
AES	Advanced Encryption Standard
CDSS	Clinical Decision Support System
CTGAN	Conditional Tabular Generative Adversarial Network
CNN	Convolutional Neural Networks
EPDP	Effective and Privacy-preserving Disease risk Prediction
FN	False Negative
FP	False Positive
GAN	Generative Adversarial Network
GBT	Gradient Tree
HDPM	Heart Disease Prediction Model
KNN	K-Nearest Neighbor
LSTM	Long Short Term Memory
MITM	Man in the Middle Attack
ML	Machine Learning
MLP	Multilayer Perceptron
NB	Naive Bayes
PPDP	Privacy-Preserving Disease Prediction
RF	Random Forest
RNN	Recurrent Neural Network
SDCA	South Denver Cardiology Associates
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TPP	Three Pass Protocol
XGBoost	eXtreme Gradient boosting

List of Symbols

Symbol	**Meaning**
θ	Theta (angle magnitude)
σ	Standard Deviation of the Distribution
$\oplus$	XOR
$\odot$	Hadamard Product

Chapter One

General Introduction

1.1 Introduction

Heart disease is a generic term for any abnormality that affects the heart, like coronary artery disease, heart failure, etc. Some heart diseases can lead to heart attacks [1]. Ordinarily, the data related to the diagnostic results and prescriptions of these diseases are saved in handwritten records. With advances in technology, these records are changed into digital forms. Patient data are therefore crucial components of modern healthcare. In addition, the privacy of such data and the patient-doctor relationship are of growing significance [2]. Consequently, this raises the issue of any illegal or unethical access to the patient's digital data. For example, over 287,000 patients' medical information has been disclosed due to a data breach at an American health clinic called South Denver Cardiology Associates (SDCA) [3].

Cryptographic methods have been proven to be effective in ensuring the secrecy of data in the presence of a malicious intruder [4]. Given a supplementary parameter called a *key*, a cryptographic method applies a mathematical formula to encrypt and decrypt sensitive messages. Based on this parameter, two categories of cryptography are distinguished: *symmetric-key* and *public-key cryptography* [5]. The work of this thesis considers applying the first category. The widely known example of symmetric-key cryptography is the Advanced Encryption Standard (AES). Currently, AES encryption is the most secure method since it employs a long key and is challenging to breach [6].

The cryptographic methods are exercised by protocols, termed cryptographic protocols. These protocols accomplish security service(s), such as secrecy, key distribution, authentication, etc. [7]. From these protocols, a Three Pass protocol (TPP) is designed to exchange confidential messages between two parties without the need for a key interchange beforehand [8].

Along with the security issues for protecting patients' data, there is a vital concern regarding the diagnosis process of heart diseases. Heart diseases are globally considered the essential causes of annual mortality of human beings [9]. As recorded by the World Health Organization, heart disorders are responsible for 17.7 million deaths each year, accounting for 31% of all fatalities. An estimated 6.7 million of these deaths were caused by asymptomatic heart attacks [10]. Thus, a silent heart attack can result in fatality and long-term heart damage. Such a mortality rate increases the awareness of reducing fatal events in the future. Indisputably, an in-time and precise diagnosis is crucial in both saving the patients' life and halting further health decline. The work of this thesis is directed at detecting whether an individual has heart disease or not, and then predicting if the heart disease patient will suffer from a heart attack in the future or not.

Artificial Intelligence (AI) techniques from the machine and deep learning branches are prevalent in medical diagnosis and prediction fields [11]. In essence, AI techniques help in decreasing human mistakes and enhancing diagnosis findings. Based on human inputs and learned datasets, machine learning algorithms reach informed decisions [12][13]. Deep learning, in contrast, employs layered algorithms to build an "artificial neural network" that can make intelligent decisions without human interventions [14]. This thesis applies techniques from both AI branches. In this work, several algorithms were used, such as K-Nearest Neighbor (KNN) [15], Random Forest (RF) [16]. After comparing the results of these algorithms, it was found that the highest results were achieved by the KNN and RF algorithms, so they were adopted. On the other hand, Long Short-Term Memory Networks (LSTMs) [17] and the

Conditional Tabular Generative Adversarial Network (CTGAN) model [18] are selected in this thesis from the deep learning branch. In essence, the LSTM is utilized as a predictor tool due to its refinement action during the prediction process, namely, the relevant information is retained while the irrelevant one is eliminated. The CTGAN is employed as a dataset expander as it generates data with statistical characteristics that are similar to the original one.

The above debate stimulates this thesis to build an automated system that pays attention to three aspects: the security of medical data, heart disease detection, and heart attack prediction.

1.2 Problem Statement

The majority of researches concentrate on the diagnosis stage of the patient's condition but neglect to consider the secrecy of the patient's profile or the medical diagnosis of the patient's condition.

AES is a commonly used symmetric-key algorithm for encrypting critical data. The AES include that the key is securely distributed between the participants and an implicit trust is confirmed in these participants.

Machine learning techniques are used to assist in enhance making classification decisions in the diagnosis process. However, choosing a machine learning technique with the most accuracy performance is still under scrutiny.

How to improve and to achieve as accurate as possible heart attack prediction value is a challenging task. Though Deep Learning networks have been widely used to predict medical events, the issue of providing perfect prediction outcomes is still progressing.

In addition, not found enough training data is a major hindrance to these networks performance.

1.3 Aims of Thesis

This thesis aims at handling the above-mentioned problems through:

1) Designing an automated system secure to diagnose heart diseases.
2) Enhancing the precision of both heart disease detection and heart attack prediction using respectively, machine and deep learning techniques.

1.4 Contributions of Thesis

1) Implementing the TPP to securely distribute a key and confidential information between participants without any prior knowledge between them and Maintaining the confidentiality of patient data by encrypting it using the AES algorithm.
2) Assessing the effectiveness of the five classification machine learning methods K-NN and RF utilizing a variety of performance metrics.
3) Conducting the CTGAN method to expand the training data set of heart patients and to improve the accuracy of the LSTM method.

1.5 Related Work

This part evaluates the literature that has been completed in related domains because the current job involves determining the presence of heart abnormalities and protecting the patient's data:

In 2018, Chuan Z. et al. proposed a secure scheme called Privacy-Preserving Disease Prediction (PPDP) which enables the cloud server to diagnose cardiovascular diseases without disclosing private information. In PPDP, firstly the medical data of patients are securely kept in the cloud by encrypting them via encryption based on matrices method, and next this data can be used by Single-Layer Perceptron machine to build prediction models. The real experiments in this work have shown that the computation costs of disease learning and prediction are lower than other schemes like Naïve

Bayesian classification, However, the limited amount of data represents the problem in this research [19].

In 2019, **Anjan Nikhil Repaka et al**. created a sophisticated algorithm for predicting cardiac illness. They used the Advanced Encryption Standard(AES) algorithm and the Naive Bayesian(NB) classification approach to create their system. They created medical files with the following characteristics so that they could be fed into the NB: Patient characteristics such as age, blood pressure, cholesterol, sex, blood sugar, and others. Data gathering, user registration, login, classification using annotations, prediction, and data transmission security using the AES algorithm are some of the steps that make up the developed system. Their approach shows that even when the attributes are limited, NB still obtains an accuracy of up to 89.77%. Moreover, AES yields better security performance outcomes in comparison to the symmetric encryption technique. The limitations in this research is Seemingly the accuracy score is relatively low as they are dealing with a medical issue. In addition, it seems to be no attention is paid by this work to the AES's key distribution problem [20].

In 2019, Xue Yang, et al., addressed the issue of security and privacy safeguards since without them, disease Risk prediction cannot advance. The EPDP system, which stands for Effective and Privacy-preserving Disease risk Prediction, is essentially a telemedicine system that they developed. It successfully completes the two stages of developing a disease model and predicting diseases while preserving privacy. During the illness model training stage, the symptom set of each disease is extracted using a combination of super-increasing series and homomorphic cryptography. During the prediction stage, the prediction results are computed using the bloom filter approach. They gave an example of how well their technology work in medical crises, However, the limited amount of data represents the problem in this research [21].

In 2019, Kuang Junwei et al. improved oblivion's internal portal input and introduced a new model. The irregular interval is first smoothed to provide a time parameter vector, which is then used as an input for the forget gate to circumvent the prediction barrier caused by the irregular interval. The efficiency of the proposed approach is supported by the experimental results, which were conducted using a set of clinical data obtained from the hospital's HIS. The dynamic prediction model proposed in this paper is superior to the traditional LSTM model in classification. An accuracy of 89% was produced by the upgraded model, However, the limited amount of data represents the problem in this research [22].

In 2019, Debjani Panda, et al., analyzed the effectiveness of several categorization algorithms under the supervision of a set of features. Feature selection is essential in reducing pointless and redundant features to decrease the cost and time of training prediction models. The classification methods that have been looked at are Logistic Regression, Random Forest, Extra Trees, and Naive Bayes. Ridge Regression and the (LASSO) operator, short for least absolute shrinkage and selection have been used to supply these algorithms with specific features. After feature selection, the classifiers' accuracy rises markedly. When utilizing Lasso regression as opposed to ridge regression, the forecast has typically increased by 33.3% instead of 30.73%. After utilizing feature selection approaches like Lasso and Ridge regression, the researchers discovered that of the four distinct classifiers utilized, the Gaussian Naive Bayes Classifier exhibits great results with an accuracy score of 94.92%, but the limitation in this research is Non-compliance with the confidentiality of information. Researchers used the Cleveland Heart dataset from the UCI database for their inquiry [23].

In 2021, Abdulaziz Albahr, et al., created a computational model to identify problems in the heart. This model includes a recommended regulator for an artificial neural network. The predictive model is incorporated into a novel regularization known as RSD-ANN, which compares the outcomes to its parents by decrementing the weights in accordance with the weight matrices' standard deviation. The proposed regulator gave the attribute coefficients high values in the weight metrics space as a punishment for performing well in the test data 96.30% accuracy was achieved with this model, However, this research did not care about maintaining the confidentiality of the information [24].

In 2021, Awais Mehmood, et al. used Convolutional Neural Networks (CNN) to detect potential heart diseases with a sophisticated dataset obtained from the UCI library. This dataset includes specific cardiac test parameters as well as normal human activities. The results showed that the proposed model performs better than the current techniques mentioned in this study. The overall accuracy of the proposed model is 97%, but It seems to be no attention is paid by this work to security concerns related to the patient's information. [25].

In 2021, MIN JONG CHEON, et al., demonstrated the potential for additional investigation into the generation of synthetic EEG data using deep learning techniques like (Tabular Generative Adversarial Network) TGAN and (Conditional Tabular Generative Adversarial Network) CTGAN. The EEG data from CTGAN exhibits higher similarity than TGAN through visualization and similarity score. The researchers attempted to use the synthesized dataset as input data for several machine learning algorithms, unlike the related research. However, this study has a problem in that machine learning models do not perform better than the real data when the synthetic data from our trials is utilized as the input data. Using data from the website www.kaggle.com, the accuracy value for all methods employed in this study ranged from 49.1% to 49.8% and the limitation in this research is when the synthetic data is used as input data for the machine Learning models, they do not appear higher performance compared to the original data [26].

In 2022, Anwar Ul Hassan, et al., published ML classifiers to predict the presence of heart problems. UCI repository was used to obtain the dataset. The collected data was cleaned and pre-processed. The goal has been reached. Then the prediction is made using ML models. Eleven ML methods used to predict heart disease were evaluated. Multilayer Perceptron(MLP) and Augmented Gradient Tree(GBT) were both used by the researchers. The results show that the Augmented Gradient Tree and Multilayer Perceptron reach an accuracy of 95% in predicting the presence of CHD from applied ML classifiers. But Random Forest(RF) managed to reach the highest classification accuracy of 96%, with a specificity and sensitivity of 96% and 95%, respectively. However, this research did not care about maintaining the confidentiality of the information [27].

In 2022, Umarani Nagavelli, et al., devised an automated technology to aid medical professionals in the early detection of heart issues. This research presents many quick analysis-based machine learning algorithms for the diagnosis of heart disease. First, weighted NB is used to predict cardiac disease. The automated analysis of the localisation of myocardial ischemia per elements from frequency, time, and information theory constitutes the second stage. The two best classifiers selected in this step for classification are Support Vector Machine (SVM) and XGBoost. A third option is an improved SVM built on the duality optimization technique, which has also been examined for its ability to detect heart failure. Finally, a Clinical Decision Support System (CDSS) employs an effective Heart Disease Prediction Model (HDPM). The analysis's findings demonstrated the XGBoost algorithm's high accuracy (95.9%), precision, recall, and F1-measure values. The precision, recall, and F1-measure values of the SVM with duality optimization are low, whereas the accuracy of the NB with a weighted approach is just 86%. However, this research did not care about maintaining the confidentiality of the information [28].

In 2022, PanelHuru H. et al. have developed a Cosine Weighted K-Nearest Neighbor algorithm for heart disease forecasting using personal data behavioral characteristics. This algorithm gleans knowledge from the Blockchain-stored data. Typically, the multi-layer Blockchain serves as a trustworthy source of educational information and a secure place to store patient data. This method have achieved 15.61% accuracy. In addition, Blockchain-based storage have attained 25.03% throughput which is greater than those of peer-to-peer storage. It is noted in this research that the accuracy is very low [29].

In 2023, Chintan M. Bhatt, et al., To classify cardiac illness, they made use of a variety of models using a real-world dataset. The k-modes clustering algorithm was used to predict the existence of cardiac disease in a dataset of patients. For both the male and female datasets, the elbow curve method was used to calculate the ideal number of clusters. The data showed an 87.23% accuracy for the MLP model. These findings demonstrate that k-modes clustering can accurately predict cardiac illness, and the technique may help in the creation of specialized therapeutic and diagnostic strategies for the condition. The 70,000-item Kaggle dataset on cardiovascular disease was used in the study. All algorithms have accuracy levels above 86%, with decision trees having the lowest accuracy(86.37%) and multilayer perceptrons having the maximum accuracy. Despite the positive outcomes, there are a few limits that should be taken into account. which this study may not apply to other demographics or patient groups since only one data set was used [30].

Table 1.1 Summary of Related Works

References	year	Dataset	Methods	Limitations	accuracy
[19]	2018	Medical data in hospital	Privacy Preserving Disease Prediction (PPDP)	the limits of data size	85%
[20]	2019	Medical profiles	• A Naive Bayesian (NB) classifier is used for the classification and prediction sides • Advanced Encryption Standard (AES) algorithm encrypts the obtained predictive data.	Seemingly the accuracy score is relatively low as we are dealing with a medical issue. In addition, it seems to be no attention is paid by this work to the AES's key distribution problem	NB = 89%
[21]	2019	UCI	Effective and Privacy Preserving Disease risk Prediction(EPDP)	the limits of data size	87%
[22]	2019	HIS	improve LSTM model	the limits of data size	89%
[23]	2019	UCI	Naive Bayes	Non-compliance with the confidentiality of information	94%
[24]	2021	University of California	RSD-ANN	Not observing information security	96%
[25]	2021	UCI	convolutional neural networks (CNN)	It seems to be no attention is paid by this work to security concerns related to the patients' information	97%
[26]	2021	www.kaggle.com	CTGAN, TGAN methods	When the synthetic data is used as input data for the machine Learning models, they do not appear higher performance compared to the original data	49%
[27]	2022	UCI	Random Forest	Not observing information security	96%
[28]	2022	UCI	XGBoost, SVM, NB	Not observing information security	XGBoost = 92%
[29]	2022	UCI	Weighted KNN	Low accuracy	15.61%
[30]	2023	www.kaggle.com	Decision tree classifier, multilayer perceptron, random forest classifier, and XGBoost	Due to the study's reliance on a single dataset, it may not be generalizable to other areas or patient groups	86%

1.6 Thesis Outline

The remaining part of the thesis is organized as the following:

Chapter Two: Theoretical Background

This chapter will provide a background of the Secure Cardiac Diagnostic System, Encryption, Classification, Prediction, and augmentation techniques used in the proposed system, and an introduction to Machine Learning and Deep Learning.

Chapter Three: The Proposed System

This chapter contains all the principles of the Secure Cardiac Diagnostic System a thorough explanation of the main stages for building it. The proposed system is explained through a flowchart that includes the detailed stages of the work, as well as an explanation of each attribute of the data set used.

Chapter Four: Experimental Results

This chapter will present the suggested system's experimental findings.

Chapter Five: Conclusions and Future Work

This chapter presents the key findings from the system's outcomes along with some recommendations for next research.

Chapter Two

Theoretical Background

2.1 Introduction

As mentioned in Chapter one, heart disease is regarded as a significant reason for silent death worldwide. The early and faultless diagnosis of heart disease is crucial to prevent further heart issues, such as heart attack. Machine and deep learning techniques can be employed to assist in precisely diagnosing this disease. In addition, there is an ongoing concern about protecting the medical data of patients, this can be addressed by the encryption methods.

This chapter provides some background information on the cryptographic primitives and cryptographic protocols. It also presents an overview of the machine and deep learning techniques that have been applied in this thesis.

2.2 Cryptographic Primitives

Cryptographic Primitives are those algorithms that employ mathematical function to conduct encryption and decryption operations on secret messages. In general, an encryption operation is a series of steps for turning an ordinary text, known as *plaintext*, into confused form, known as *ciphertext*. Whereas the operation of retrieving the original *plaintext* from the *ciphertext* is called decryption. Both of these operations rely on a piece of data, called *key*, which allows only the parties that hold it to perform these operations on a message [31].

Depending on the number of keys using for cryptographic operations, the following cryptographic primitives are recognized:

2.2.1 Public-Key Cryptography

Public-Key Cryptography, also called asymmetric key cryptography, is a cryptography entailing that no secret key is distributed between the involved participants. Generally, the public-key cryptography uses a couple of keys, a publicly known key (called public key), a secretly known key to only its owner (namely private key). As this type of cryptography is not of our concern, therefore it will not be described deeply in this chapter [32].

2.2.2 Symmetric-Key Cryptography

Symmetric-Key Cryptography is a classical cryptography at which the key for both the encryption and decryption processes is identical. The algorithms for this cryptography type necessitates that the key is distributed between the involved parties in advance. The widespread symmetric key algorithms are: Data Encryption Standard (DES), Advanced Encryption Standard (AES), and Rivets Cipher 4(RC4) [33]. As the AES algorithm has been applied in this thesis, it will be described in detail as follows:

2.3 Advanced Encryption Standard

Advanced Encryption Standard(AES) is one of the common symmetric-key encryption paradigm [34]. AES is an iterative cipher that is based on two operations: "substitution" and "permutation". Essentially, it consists of a number of interconnected processes, some of which require swapping inputs for particular outputs(substitutions), while others involve randomizing bits (permutations) [35]. AES never use bits for its calculation; instead, it uses bytes. Therefore, AES treats a plaintext block of 128 bits as 16 bytes. For processing as a matrix, these 16 bytes are organized into four columns and four rows [36].

It uses keys with lengths of 128, 192, and 256 bits to encrypt plain text divided into blocks of 128 bits. The number of rounds in AES is configurable and is based on the size of the key. For 128-bit keys, 192-bit keys, and 256-bit keys, AES uses 10, 12, and 14 rounds, respectively. Four stages, namely: **s**ubstitution of the bytes, shifting the rows, mixing the columns, and **a**dding the round key, make up each round of AES, except for the last round that excludes the mix columns stage in encryption process, and also the inverse mix columns stage is omitted in the decryption process, see Figure (2.1)[37].

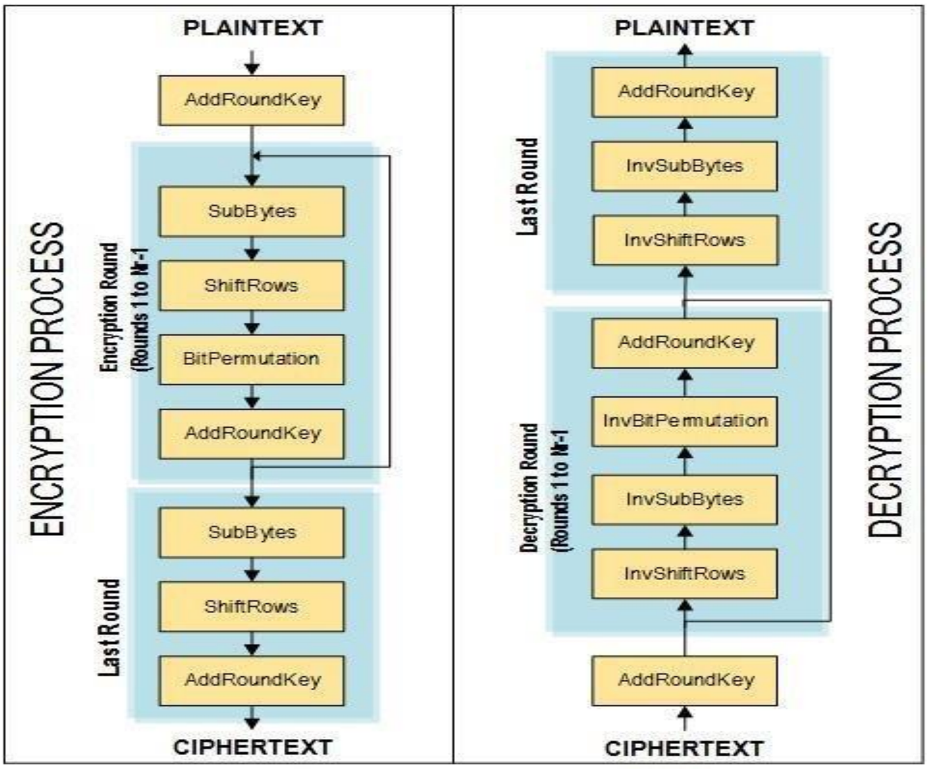

Figure 2.1 A General Process of AES Algorithm [37]

In the adding round key stage, the 128 bit key is initially xored with plain text. The 128 bit plaintext block is partitioned into 16 bytes. These bytes are represented as a 4 x 4 matrix called the *State*. Every single State byte is transformed into a new byte that is created by intersecting row and column elements. Inverse State is used for substituting byte transformation during decryption. During shifting the rows stage, the first row of the State matrix remains the same, whereas the second, third, and fourth row experiences a one-byte circular left shift, a two-byte circular left shift, and a three-byte circular left shift, respectively. The matrix multiplication of the state is used to carry out mixing the columns stage. The expanded key is xored with the state in the adding the round key stage to obtain the ciphered message [37].

2.4 Cryptographic Protocols

Cryptographic or security protocols are pre-determined and disseminated procedures that employs cryptographic algorithms to fulfill security service(s), like data confidentiality, in a mistrustful environment. Formally, a protocol is a limited sequence of steps for transferring messages, each of which has the following structure:

$$i. A_i \rightarrow B_i: M_i \quad \ldots\ldots\ldots\ldots (2.1)$$

Where $1 \leq i \leq n$, is the i^th step of a protocol, n is an overall number of the protocol's steps, $A_i$and $B_i$represent the sender and the receiver, respectively, of step i, and M_i is the message of step$i$. This form represents that M is sent from A to B. There are different examples of the cryptographic protocols, but we will state only one example below, which has been utilized in our thesis [38].

2.4.1 Three-Pass Protocol

The Three-Pass protocol (TPP) was created to transmit a private message over an unsecure communication network without the need to share or pass any secret key. As suggested by its name, this protocol is intended to exchange the following three messages [8]:

$$1. A \rightarrow B: \{m\}_{K(A)}$$

$$2. B \rightarrow A: \{m\}_{K(A)}\}_{K(B)}$$

$$3. A \rightarrow B: \{m\}_{K(B)}$$

In the first message, A uses its private key to send B an encrypted message. Then B re-encrypts the received message under its secret key and transmits the double encrypted message back to A. Finally, A decrypts the received message and transmits B the decryption outcome. B receives the third message, decrypts it, and discovers the hidden message [8].

The encryption and decryption processes in this protocol are carried out by Xoring the message and the key.

2.4.2 Protocol Attacks

A protocol run that meets an unwanted property, such as compromising authentication or disclosing a secret, is referred to as an attack on the protocol. A dishonest player with identity I known as intruder(or attacker) who launches an attack on a protocol. The $I(X)$ notation denotes an imposter or intruder who acts in place of the legitimate participant with identity X.

While there are different skillful attacks that can be launched by the imposter participant, only the following attacks, that have been applied in this thesis, will be illustrated [39].

1) **Man-in-the-Middle (MITM) Attack**: This type of attack involves a third party furtively intercepting, replaying, and changing messages between two original participants who continue to believe they are the only ones interacting. In this attack, a third party is present between the two parties, see Figure (2.2) [39].

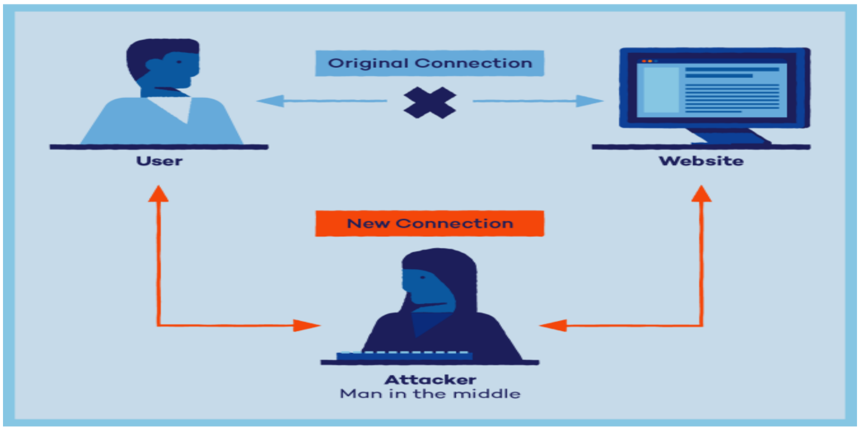

Figure 2.2 The Main Concept of Man-In-The-Middle Attack [39]

a) **Dictionary Attack:** this attack is launched depending on two strategies:
1) Compromising the user's password by trying each actual word found in the English dictionary as a password.

2) Computing the possible hash of each actual word found in the English dictionary before checking the password list for matching [39], see Figure (2.3) .

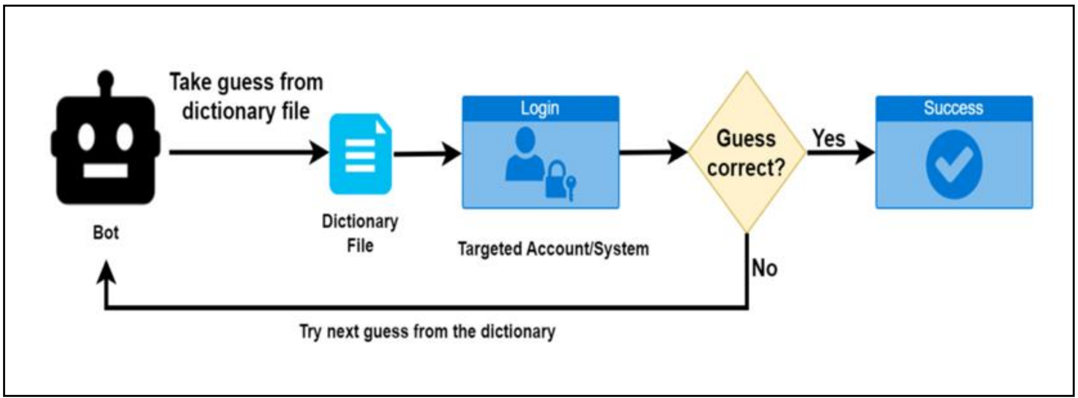

Figure 2.3 The Main Concept of Dictionary Attack [40]

2.5 Cardiology Diagnostic Systems

The automated method of detecting heart diseases is based on a set of biomarkers such as diabetes, high blood pressure, anemia, and others called heart disease diagnostic systems. The severity of cardiac disease in humans has been determined using various data mining and neural network techniques. HD is frequently identified by a clinician following a review of the patient's medical history, the findings of their physical examination, and any alarming symptoms. However, this method of diagnosis does not consistently identify patients with heart disease. The analysis is also expensive and computationally difficult. A non-invasive diagnosis system based on machine learning (ML) classifiers must be developed to address these issues [41].

2.6 Classification

Classification is one type of predictive method. In a specific way, it can be defined as arranging or sorting objects into groups on the basis of a common property that they have [42]. This can be seen as the application most used in data mining techniques that utilizes a set of instances classified in advance to improve a model that can classify the set of all records. These methods often use a decision tree or otherwise a set of classifying algorithms that are based on neural networks. The data classification process includes learning and classification. During the act of learning, a testing of the training data is performed by the classification algorithm. This data is used throughout the test to evaluate the extent to which the classification outcomes are accurate. This turns out to be reasonable, and this would mean that these rules can be used for data records that follow. Its application in detecting fraud will involve full records of fraud as well as valid actions which are defined based on so-called record-by-record. The algorithm for classification training adapts such instances that have been classified in advance to define the parameters required for its appropriate discrimination. These are later encoded into a classifier model using the algorithm itself. One of the problems faced during the procedure of classification is model construction [43]. The categorization process involves building models, which is a supervised learning problem. The description of the training instances is presented in terms of two factors:

(1) features, these are either categorical, numeric values or symbols in any order.

(2) class label, also referred to as the anticipated or outcome feature.

Classification of algorithms

- Classification by decision tree induction
- Bayesian Classification
- Neural Networks
- Support Vector Machines (SVM)
- Classification Based on Associations

- Classification by K-Nearest Neighbor
- Classification by Random Forest

2.6.1 Machine Learning Models:

A branch of artificial intelligence (AI) that enables computer programs to learn from data and then make appropriate decisions based on the information that has been learned from prior experience. It depends on computer science, statistics, and mathematics [44].

There are a variety of classification methods available, such as K_Nearest Neighbor (KNN) and Random Forest (RF):

1) The K_Nearest Neighbor:

KNN is a well-liked machine learning method that is typically used for the classification of benchmark data sets. Even when compared to the most sophisticated machine-learning techniques, this straightforward and user-friendly algorithm produces good results across many fields. The growing accessibility of data given in novel formats, such as free text, photos, music, and video, has prompted recent interest in KNN. However, the choice of the distance metric and the k parameter by the following equation are the two most important variables that affect how well KNN performs: [46]

Distance (x1,x2),(y1,y2) = $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ (2.2)

The work of the algorithm can be explained in more details through Figure (2.4).

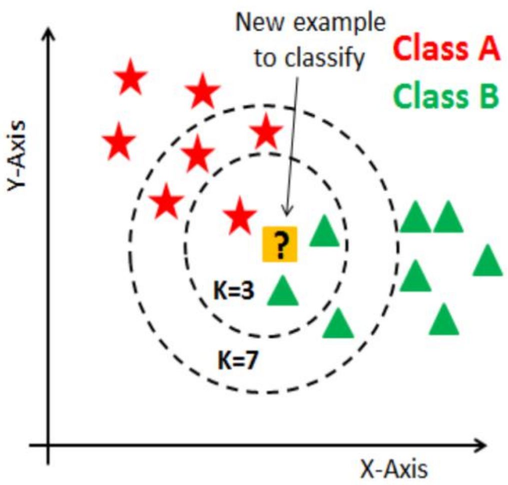

Figure 2.4 KNN Algorithm Structure [47]

Figure (2.4) shows the steps of the KNN algorithm, which are shown below:

1- Find the value of the k variable, which represents the quantity of neighbors.

2- Determine how far apart the new example is from the examples in the dataset.

3- Depending on the minimal distance that was determined in the previous stage, arrange the examples to get the adjacent ones, and count the number of k neighboring examples from them.

4- Define the class for the neighbors.

5- The class that has the majority of the neighbors is the expected class for this example [48].

Algorithm 1: KNN [48].

Algorithm 1: ***K-Nearest Neighbor Classifier***
Input: V_{ij}, *where* $i = 1,2,\ldots,x$, *and* $j = 1,\ldots,y$, *x represents the feature values, where y is the number of values associated with each feature, n is the total number of features.*
Output:: *Classified feature values* $ClTV_{lj}$ *are converted into* Cl_l *class labels, such that* $l = 1,0$, *and* Cl_1 *is a person suffering from a heart condition, while* Cl_0 *is denotes a healthy person*
Begin: ***1:*** *Think of* RV_{ij} *as the training set samples, chosen at random from the input, and* TV_{ij} *as testing set samples, also chosen at random.*   ***2:*** *Verify and categorize the training set* RV_{ij} *into* Cl_l, *and save the results in* $ClRV_{lij}$.

3: *Choose TV_j from TV_{ij} to represent a value that needs to be classified.*

4: *Calculate Euclidean distance between TV_j and RV_{ij} using the following equation:*

$For\ i = 1\ to\ x$

$For\ j = 1\ to\ y$

$$d(TV_j, RV_{ij}) = \sqrt{\sum_{j=1}^{y} (RV_{ij} - TV_{ij})^2}$$

5*: Sort the computed Euclidean distances in a non-decreasing order.*

6*: Select how many neighbors you want, then save it in k.*

7: Choose k values from the sorted distances and enter them. $d(TV_j, RV_{ij})_k$.

8*: Determine the class of.* TV_j

$no_{Cl_1} = 0, no_{Cl_2} = 0$

$For\ p1 = 1\ to\ k$

If RV_{ij} *(which ∈* $d(TV_j, RV_{ij})_k$*) has* $ClRV_{1ij}$*Then*

$no_{Cl_1} = no_{Cl_1} + 1$

Else

$no_{Cl_2} = no_{Cl_2} + 1$

End if

End For

If $no_{Cl_1} > no_{Cl_2}$ ***Then***

Put TV_j *in* $ClTV_{1j}$

Else

Put TV_j *in* $ClTV_{2j}$

End if

2) The Random Forest:

Because the model belongs to the classification family, it is sometimes referred to as the supervised learning algorithm. In this model during the learning stage, the first thing that is made is a forest, or a grouping of numerous random trees. For instance, a dataset with x attributes randomly selects a feature called y in the beginning. It generates nodes while employing all features and the best rift technique. The technique can also be used to build a full forest by repeatedly doing the previous steps. The program then attempts to join the trees using the anticipated outcome and voting procedure during the prediction process. To exclude the tree with the greatest forecast, the random trees are combined through voting in a forest. This can improve the predictive accuracy for future data [49].

You should be aware that you frequently employ the Gini index, or the formula used to determine how nodes on a decision tree branch are ordered when executing Random Forests based on categorization data.

$$Gini = 1 - \sum_{i=1}^{n} (Pi)^2 \qquad (2.3)\ [49]$$

Based on the class and likelihood, this formula determines the Gini of each branch on a node, showing which branch is more likely to occur. Here, pi denotes the class's proportional frequency throughout the dataset, while n is the overall number of classes [49].

Figure (2.5) provides a more thorough explanation of how the algorithm operates.

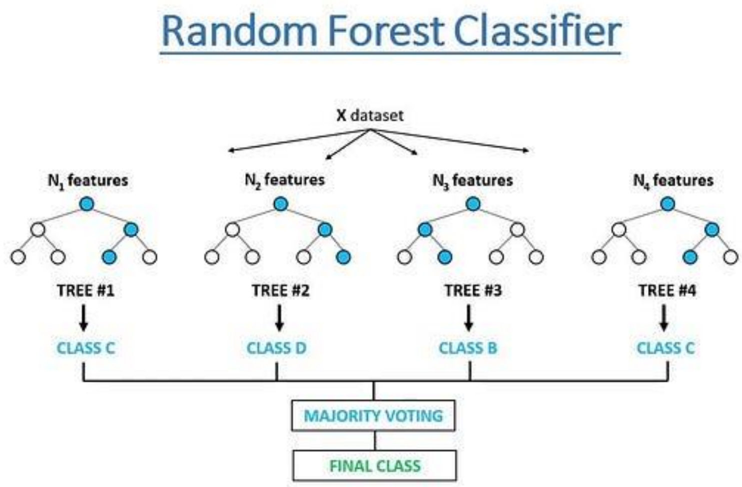

Figure 2.5 RF Algorithm Structure [50]

Figure (2.5) shows the structure of the RF algorithm, and is outlined in the steps below:

1- Select random sample data points from the training set.
2- For each training set of data, this algorithm will build a decision tree.
3- The choice tree will be averaged during voting.
4- Finally, choose the prediction result that received the most votes as the final forecast result [51].

Algorithm 2: RF [51]

Algorithm 2: Random Forest Classifier
Input: V_{ij}, *where i=1,...,x and j=1,...,y for the feature stand, x for the number of features, y for the number of values related to each feature, no_trees for the number of trees assumed, and Fl for the vector of feature labels.*
Output: *The classified feature values* $ClTV_{lj}$ *are converted into* Cl_l *class labels, such that* $l =$ 1,0, *and* Cl_1 *is a person suffering from a heart condition, while* Cl_0 *is denotes a healthy person*

Begin:
1: Think of RV_{ij} *as the training set samples, chosen at random from the input, and* TV_{ij} *as testing set samples, also chosen at random.*
2: Verify and categorize the training set RV_{ij} *into* Cl_l, *and save the results in* $ClRV_{lij}$.
3: Choose TV_j *from* TV_{ij} *to represent a value that needs to be classified.*
4: Make a forest
For $p1 = 1$ *to* no_{trees}
Determine how many features make up the subset $no_{sub-features} < x$, $no_{sub-features} = \sqrt{x}$
randomly divide the labels Fl feature into $no_{sub-features}$ *subsets, and assign them to* $SFl_{no_{trees},no_{sub-features}}$
For $q1 = 1$ *to* $no_{sub-features}$
random selection of values for $SFl_{p1,q1}$ *from* $RV_{p1,q1}$ *to be a dataset of randomness* $Y_{p1,q1}$ *Create a tree using* $SFl_{p1,q1}$ *and* $Y_{p1,q1}$
End For
Use $SFl_{p1,q1}$ *and* $Y_{p1,q1}$*to make a decision* (Cl_l) *related to each tree that has been built.*
End For
5: *Identify the category of* TV_j
For *each column in* TV_{ij}
For $p1 = 1$ *to* no_{trees}
If *a column that matches the created tree* ***Then***
Calculate the majority of the choices made in relation to the built-in trees.
End if
End For
$ClTV_{lj} =$ *the most of decisions are deliberate*
End For

2.7 Deep Learning (Deep Learning and Deep Neural Networks)

One way to define deep learning is as a class of machine learning techniques that utilizes multiple non-linear data performance layers for either supervised or unsupervised data reduction, feature extraction, feature transformation, pattern discovery, and classification. Another approach to describe this idea is as a subfield of machine learning that relies on multi-level representation techniques to model intricate relationships between pieces of information. Bringing machine learning closer to its original goal of artificial intelligence is a recent area of research. Learning several layers of representing and abstracting utilized to make sense of data such as an image, sound, and text is the main focus of this idea [52]. According to what was previously discussed, the two common features of the deep learning definitions are:

1) having a foundation of numerous computational layers or stages used to simulate nonlinear information processes and abstractly identify the dataset;

2) being utilized to extract features from the dataset using supervised and unsupervised algorithm learning. This idea has been used in a variety of fields, including speech recognition, drug discovery, and image categorization [53]. These have been accomplished through the use of several of straightforward nonlinear units that transform the original representation of raw data into more intricate functions. Consequently, these might be trained and then used to do tasks like categorizing, forecasting, and identifying data [54]. In addition, because deep learning techniques use a bigger data set, they enable modern computers to do more sophisticated neural network computations more quickly and with less risk of over-fitting.

2.7.1 Recurrent Neural Networks (RNNs)

RNNs were introduced in the 1980s, and their popularity has increased recently due to the intellectual development and recent technologies that made them capable of training. RNNs differ from feed-forward networks because they benefit from a special type of neural layer, called recurrent layers, which make the network maintain the state between the uses of the network [55]. Signals are propagated forwards as well as backward directions in the RNN to provide dynamic network memories. Apart from the data in the current time stage and the feedback, the process could make a time delay module providing internal input data to the next time step. RNNs can find the valid hidden dynamic memories of nonlinear systems [56] Figure (2.6) illustrates the RNN structure. RNN is a distinct sort of ANN network which uses serialized information because of communications directed between individual layer units. RNN is capable of storing memory because the current output depends on the previous calculations. However, it is known that RNNs revert to only a few steps because of suffering from vanishing and exploding gradient problems [57].

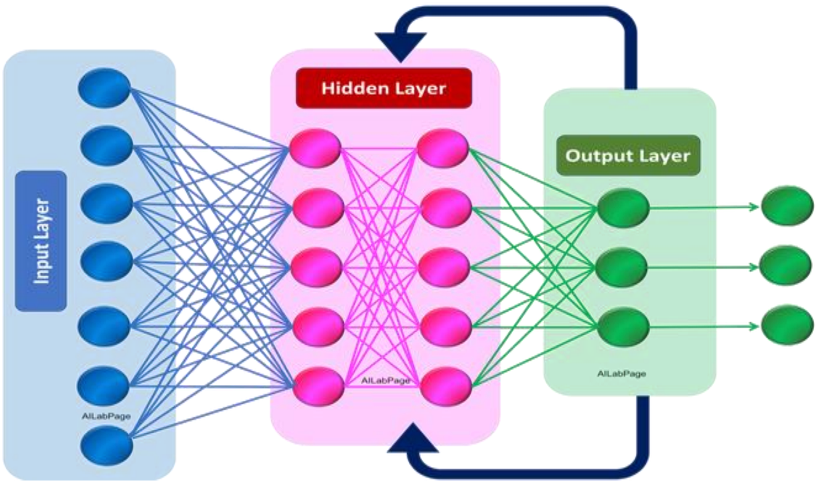

Figure 2.6 Deep RNN Structure [58]

RNNs are facing the challenges of vanishing and exploiting the gradient problem. Using sigmoid activation functions in the training of RNNs based on gradient descent can be complicated. In the exploding gradients, with the backpropagation of long-term gradients over time, there are two possible consequences: there will be either a continual growth that leads to an explosion, or a decrease that reaches null and vanishes. There are many methods proposed to cope with and solve the problem of data that explodes or vanishes, including but not limited to the use of second-order derivative as training, to preserve the magnitude of RNN long-term gradients using the rectified linear unit with sigmoid activation functions that can be training its amplitudes [59]. Also, Long-short-term-memory (LSTM) is specifically created in such a way to cope with the RNNs problems where it introduces new gates that can apply a more accurate control for the gradient. LSTM will be explained in detail in the next chapter.

2.7.2 Long Short-Term Memory

A special case of RNNs is LSTM. It is made to deal with the issue of gradients that burst or disappear. The memory cell and several gates make up the LSTM's fundamental design. In [17] , these memory cells and gates were initially introduced and are now added to every neuron in the network. The fundamental idea behind LSTM work is reliable information transmission through several of time steps to the following time step. Figure (2.7) depicts a memory circuit with an LSTM cell that uses the model's gates along with other parameters for long-term recollection or storage of data from the recurrent layer.

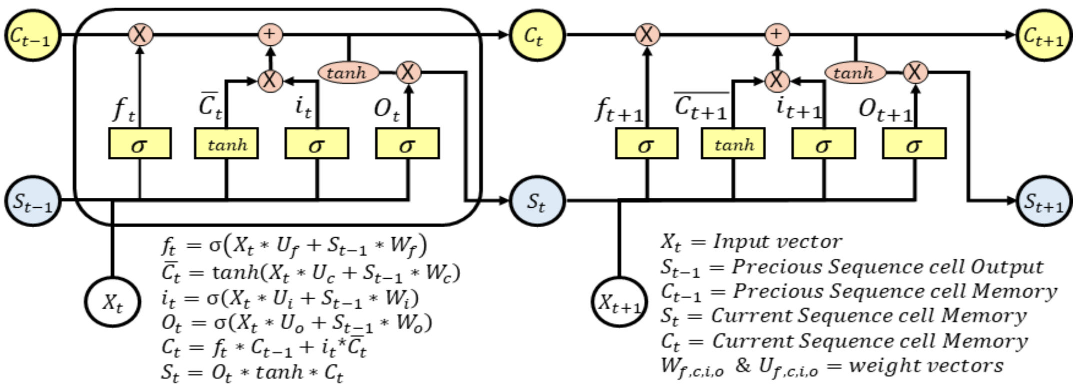

Figure 2.7 Long Short-Term Memory Neural Network Algorithm [60]

In the diagram above, the input gate is represented by i, the forget gate by f, the output gate by o, the candidate hidden state by $\bar{C}$, the internal memory of the unit by c_t, and the hidden state by s_t. The sigmoid function is used to do an element-wise calculation for each element of the input vector using i, f, and o as gates, which are transformed into a value between 0 and 1. To determine how much of the current input value has been transferred, the input gate modifies the information transfer rate. The input value's long-term or short-term memory status is adjusted by the forget gate. To determine how much state information is output, the output gate modifies the information transfer rate. The activation function tanh and the pre-existing state are used to generate the candidate output value, which is displayed by the candidate hidden state($\bar{C}$). Through gate calculation, LSTM only supplies a portion of the candidate output value as the output, as opposed to the candidate output value completely. The value c_t is created by averaging the values computed for each component of the hidden gate value, each candidate state component, and each input gate component. It stands for the fusion of recent input and previous memory. By computing the ct value and each output gate component, s_t determines the final output value [60].

The gates are introduced to the recurrent function f to deal with problems that are exploding or disappearing. The following is how the LSTM cells are used:

$$i_t = \sigma(w_i.[h_{t-1}, x_t] + b_i) \qquad (2.4)$$

$$f_t = \sigma(w_f.[h_{t-1}, x_t] + b_f) \qquad (2.5)$$

$$o_t = \sigma(w_o.[h_{t-1}, x_t] + b_c) \qquad (2.6)$$

$$\tilde{C} = \tanh(h_{t-1}, x_t] + b_c) \qquad (2.7)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \qquad (2.8)$$

$$h_t = o_t \odot \tanh(C_t) \qquad (2.9)$$

In the above equations:

i_t is the input gate.

f_t is the forgate gate.

o_t is the output gate.

While W and b stand in for the LSTM's parameters. C_t and C, respectively, represent the old and new potential values for the cell state.

In (i_t, f_t, and o_t), the sigmoid function is utilized three times. These gates' output ranges from 0 to 1, as seen in equations (2.4),(2.5),(2.6)[6]. The three gates' decisions are influenced by the prior output, h_(t-1), and the current input, x_t. The quantity of past state is let through by the forget gate f_t. The input gate determines whether the most recent input data requires an update or addition to the cell state. Depending on the state of the cell, the output gate decides what data should be output. To learn and store knowledge about the long and short-term series, these two gates work together. The LSTM's work revolves around the cell state. A specific kind of conveyor belt is the cell state. The memory cell C work as a stack of state data. Equation (2.8) is used to update the old cell state C_{t-1} to the new state of cell C_t. Equation (2.7) is used to compute the new candidate values $\tilde{C}$of the memory cell. While equation (2.9) is used to find the output of the current block of LSTM by applying the hyperbolic tangent function. The propagation of the state vector can be observed, and its interactions are linear over time. Then, the result is the gradient that connects inputs to several time steps in the past with the current output which does not significantly weaken as in the RNN. Weights and biases are learned from the network by reducing the error as much as possible between the network output and the desired training samples. This process of LSTM cell is repeated for all the training samples. The steps of LSTM can be illustrated by Algorithm 3[60].

Algorithm 3: The Long Short Term Memory

D represents the number of memory (LSTM block)
S_j represents number of cell in block j
Input: $x \longleftarrow [x_1, \dots, x_n], x_t \in {}^{m}$

Output: $h \longleftarrow [h_1, \dots, h_n], h_t \in {}^{p}$

Begin:
1. Given parameters $W_{fjm}, b_{fjm}, W_{cjm}, b_{cjm}, W_{ijm}, b_{ijm}, W_{ojm}, b_{ojm}$
2. Initialized $h_0, c_0 = \vec{0}$ $of\ length\ p$
3. *for* $\forall\, j \in D$ *do*
4. *for* $\forall\, v \in S_j$ *do*
5. $f_{tj} \longleftarrow \sigma(W_{fjm}.\,[h_{t-1}, x_t] + b_f)$
6. $C_{tj} \longleftarrow \tanh(w_{cjm}.\,[h_{t-1}, x_t] + b_c)$
7. $i_{tj} \longleftarrow \sigma(W_{ijm}.\,[h_{t-1}, x_t] + b_i)$
8. $C_{tj} \longleftarrow f_{tj} * C_{tj-1} + i_{tj} * C_{tj}$ //Update cell state//
9. $O_{tj} \longleftarrow \sigma(W_{ojm}.\,[h_{tj-1}, x_{tj}] + b_o)$ //calculate Output
10. $h_{tj} \longleftarrow O_{tj} * \tanh C_{tj}$
11. End for
12. End for
13. // Backward pass in LSTM //
14. *for* $\forall\, j \in D$ *do*
15. *for* $\forall\, v \in S_j$ *do*
16. $\Delta w_{ojm}(t) \longleftarrow \alpha\,\delta_{oj}(t) - x_m(t)$ \\ weight update of output gate\\
17. $\Delta w_{ijm}(t) \longleftarrow \alpha \sum_{v=1}^{Sj} e_s\ c_j^v(t)\ \frac{\partial sc_j^v(t)}{\partial w_{ijm}}$ \\ weight update of input gate\\
18. $\Delta w_{fjm}(t) \longleftarrow \alpha \sum_{v=1}^{Sj} e_s\ c_j^v(t)\ \frac{\partial sc_j^v(t)}{\partial w_{fjm}}$ \\ weight update of forgate gate\\
19. $\Delta w_{cjm}(t) \longleftarrow \alpha \sum_{v=1}^{Sj} e_s\ c_j^v(t)\ \frac{\partial sc_j^v(t)}{\partial w_{cjm}}$ \\ weight update of state cell\\
20. End for
21. End for
Return $h \longleftarrow [h_1, \dots, h_n], h_t \in {}^{p}$
End algorithm

2.8 Data Augmentation

A variety of methods collectively referred to as "data augmentation" can be used to artificially increase the amount of data by producing more data points from current data. This involves employing deep learning models to add new data points or make a few simple changes to the existing data. Data augmentation improves machine learning models' performance and output by adding more unique examples to training datasets. Large and sufficient datasets improve the performance and accuracy of machine learning models. It might take a lot of time and money to gather and label data for machine learning models. Businesses can change datasets to reduce these operational costs by employing data augmentation techniques. There are numerous GAN-based techniques for creating artificial tabular data, including: Vanilla GAN, Conditional GAN (CGAN), Deep Convolutional GAN(DCGAN), CycleGAN, Generative Adversarial Text to Image Synthesis, Style GAN, Super Resolution GAN(SRGAN) [61].

2.9 Conditional Tabular Generative Adversarial Network (CTGAN)

CTGAN utilizes a generative adversarial network (GAN) to model the distribution of tabular data and selects representative rows from that distribution. To address CTGAN's non-Gaussian and multimodal distribution, A GAN is made up of two neural networks that compete with one another: the generator, which creates fake data, and the discriminator, which is taught to determine whether the input is real or fake. The generator learns to produce false data that is in distinguishable from genuine data during the training process to trick the discriminator. As a result, the resulting GAN might be used to build a synthetic dataset that is wholly false but nonetheless has the same structure as real data [62]. CTGAN simulates records one at a time. It begins by choosing one of the features at random. Then, it chooses a value at random for that variable. To deal with the unbalanced categorical columns, CTGAN employs a conditional generator and training-by-sampling. Figure (2.8), which displays histograms of both genuine and fake data, serves as an example of how CTGAN's job is done [63].

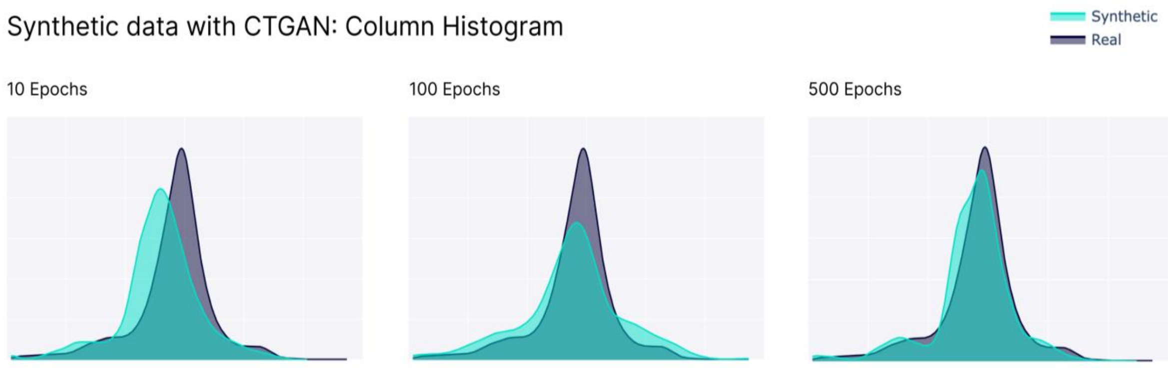

Figure 2.8 CTGAN Schema [63]

The generator and discriminator networks engage in a game during the training of a GAN. Let us first introduce the general framework of generative adversarial networks. The training of a GAN is a game between two competing networks: the generator and the discriminator. The generator G is a neural net with parameter vector θ_g that takes in argument a vector of random noise Z with distribution F_z, and maps it to the space of the data we wish to model. Usually, the components of the vector Z are independent standard Gaussian random variables, and the dimension of Z is lower than the that of the data. The resulting $G(Z; \theta_g)$ is a fake data point, and its distribution is denoted by F_g. The goal of the training procedure is therefore to find a good approximation F_g of the unknown distribution of a true data point X, denoted F_x. To achieve this goal, a competing network, the discriminator D with parameter vector θ_d, learns to determine whether a data point is real or fake. To this end, the parameters θ_d of D are trained to maximize the expected score of a real data point $EX\{D(X;\theta_d)\}$ and to minimize the expected score of a synthetic data point $EZ[D\{G(Z;\theta_g); \theta_d\}]$. To achieve the goal of generating realistic data points, the parameters θ_g of the generator are trained to maximize the discriminator's score on a fake data point $EZ[D\{G(Z;\theta_g);\theta_d\}]$. Combining the two problems together, the two networks aim to solve.

$$\min_{\theta_g} \max_{\theta_d} E_x [\log\{D(X;\theta_d)\}] + E_z[\log[1 - D\{G(Z;\theta_g);\theta_d\}]] \qquad (2.10)\ [64]$$

To deal with the non-Gaussian and multimodal distribution of continuous data, CTGAN additionally employs mode-specific normalization [64].

2.9.1 Metrics for CTGAN Performance Evaluation

There are many metrics by which CTGAN performance is evaluated, such as Memory Requirements, Machine Learning Efficacy, Statistical Similarity, Distance to Closest Record(DCR), Categorical CAP(CCAP), Numerical LR(NLR), and NewRowSynthesis. This metric (NewRowSynthesis) measures whether each row in the synthetic data is new, or whether it exactly matches an original row in the real data. This metric also looks for matches in missing values. It ignores any other columns that may be present in the data. It also searches for matching rows between the real and synthetic datasets. To be considered a match, all the individual values in the real row must match the synthetic row. The exact matching criteria are based on the type of data where this metric measures every value in the real and synthetic data(x) [65]. This is shown in the formula below, where r represents all the values in the real data.

$$\text{scaled}(x) = \frac{x - \min(r)}{\max(r) - \min(r)} \qquad (2.11)\ [65]$$

Finally, They will compute the proportion of rows in the synthetic data that match a row in the real data. The score is the complement, ensuring that **1** means The rows in the synthetic data are all new, there are no matches with the real data(good score) while **0** means All the rows in the synthetic data are copies of rows in the real data(the worst score) [65].

$$\text{score} = 1 - \frac{\text{matchingsyntheticrows}}{\text{totalsyntheticrows}} \qquad (2.12)\ [65]$$

2.10 Performance Measures

To determine how well a machine learning model is performing in its predictions, the process of constructing the model must include measuring the accuracy of the model. The evaluation metrics are affected by the problem type. Once the model has been developed, it may be assessed by looking at the prediction error rates. The errors show how many incorrect predictions the model makes. The fundamental idea behind accuracy assessment is contrasting the original objective with the anticipated one [66].

In this thesis, the following metrics are utilized:

1- ***Avalanche Effect Criterion**:* This criterion states that a considerable change should be resulted in the ciphered text, when a slight change is carried out on the input text [67]. It can be calculated through the formula below:

$$Avalanche\ Effect = \frac{no.of\ changed\ bits\ in\ the\ plain\ text}{no.of\ bits\ in\ plain\ text} \qquad (2.13)\ [67]$$

2- ***Accuracy***: the number of accurately identified cases, whether positive or negative, serves as a measure of accuracy [68].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (2.14)\ [68]$$

3- ***Precision***: testing the true positive from the Expected positives yields information on the accuracy of the model's performance [68].

$$Precision = \frac{TP}{TP+FP} \quad (2.15)\ [68]$$

4- ***Recall***: Recall is the accuracy with which positive samples are correctly identified [68].

$$Recall = \frac{TP}{TP+FN} \quad (2.16)\ [68]$$

5- $\boldsymbol{F_{measure}}$: for calculating a balanced mean output, the F1-score demonstrates the combination of precision and recall [68].

$$F_{measure} = \frac{2*Recall*Precision}{Recall+Precision} \quad (2.17)\ [68]$$

6- ***Standardization:*** results in a distribution that has a mean of 0 and a variance of 1.

$$V_{new} = \frac{V_i - V_\mu}{V_\sigma} \quad (2.18)\ [69]$$

Where V_i is a dataset's feature value, V_{new} is a scaled value for a feature, V_μ is the feature values' mean, and V_σ is the feature value standard deviation [69].

7- ***Confusion Matrix:*** presents a table arrangement of the various outcomes of the prediction and discoveries to aid in seeing the outcomes of a categorization problem. It shows a table with all of the predicted and actual values from a classifier(as shown in Figure(2.9)) [70].

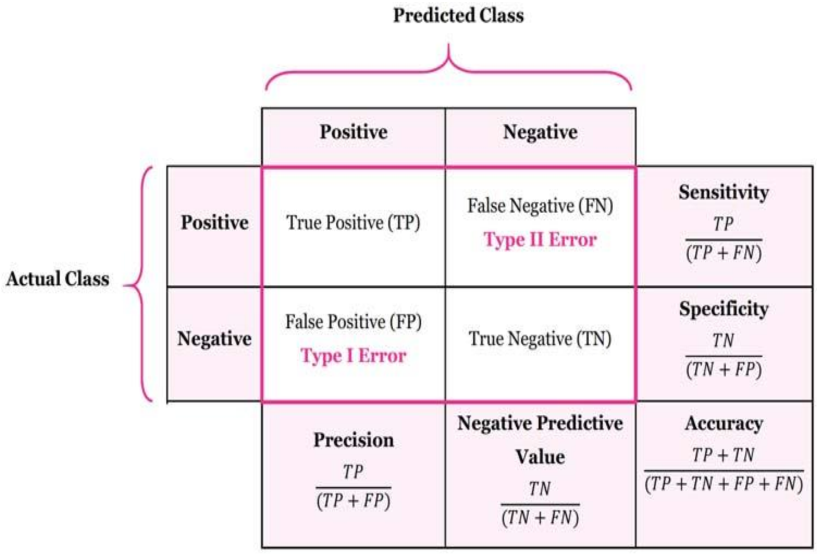

Figure 2.9 Confusion Matrix [70]

The prediction error is recorded by four parameters:

- **True Positive (TP)** is the accurate classification of the positive states as positive states.
- **False Positive (FP)** signifies the bad conditions that are erroneously classified as good conditions.
- **True Negative (TN)** is the appropriate classification for a negative diagnosis.
- **False Negative (FN)** determines the occurrences of positivity that are wrongly classified as negative [71].

Chapter Three

The Proposed System

3.1 Introduction

This chapter introduces a detailed description of our proposed system, which is intended to ensure the secrecy of medical data while reliably identifying heart disease and predicting heart attacks. Essentially, this chapter details the main structure of our proposed system, the three stages that constitute this system and also includes a detailed description of the data sets used in our system.

3.2 Structure of the Proposed System

This research suggests a secure heart disease diagnosis system that benefits the health sector by providing triple services: safeguarding the personal and medical data of a patient, assuring precise heart disease diagnosis and accurate heart attacks anticipation. Typically, the system has been developed to tackle three issues, that are mentioned in Chapter 1 namely, the key distribution problem associated with symmetric-key encryption schemes, selecting an accurate heart disease classifier, and expanding the trained dataset to increase the prediction's accuracy.

Figure 3.1 depicts an overview of our system, that includes the following stages:

1) **Security Provision stage:** This stage employs the Three Pass Protocol for exchanging the AES key in order to achieve dual authentication, in which the user and the sent data are both authenticated.
2) **Patient State Detection stage:** This stage exercises KNN and RF classifiers to correctly determine whether or not a someone has heart disease.
3) **Heart Attack Prediction stage:** This stage utilizes the LSTM predictor to faultlessly decide if or not a diagnosed person with heart disease will have a heart attack.

The below subsections will explicate these stages.

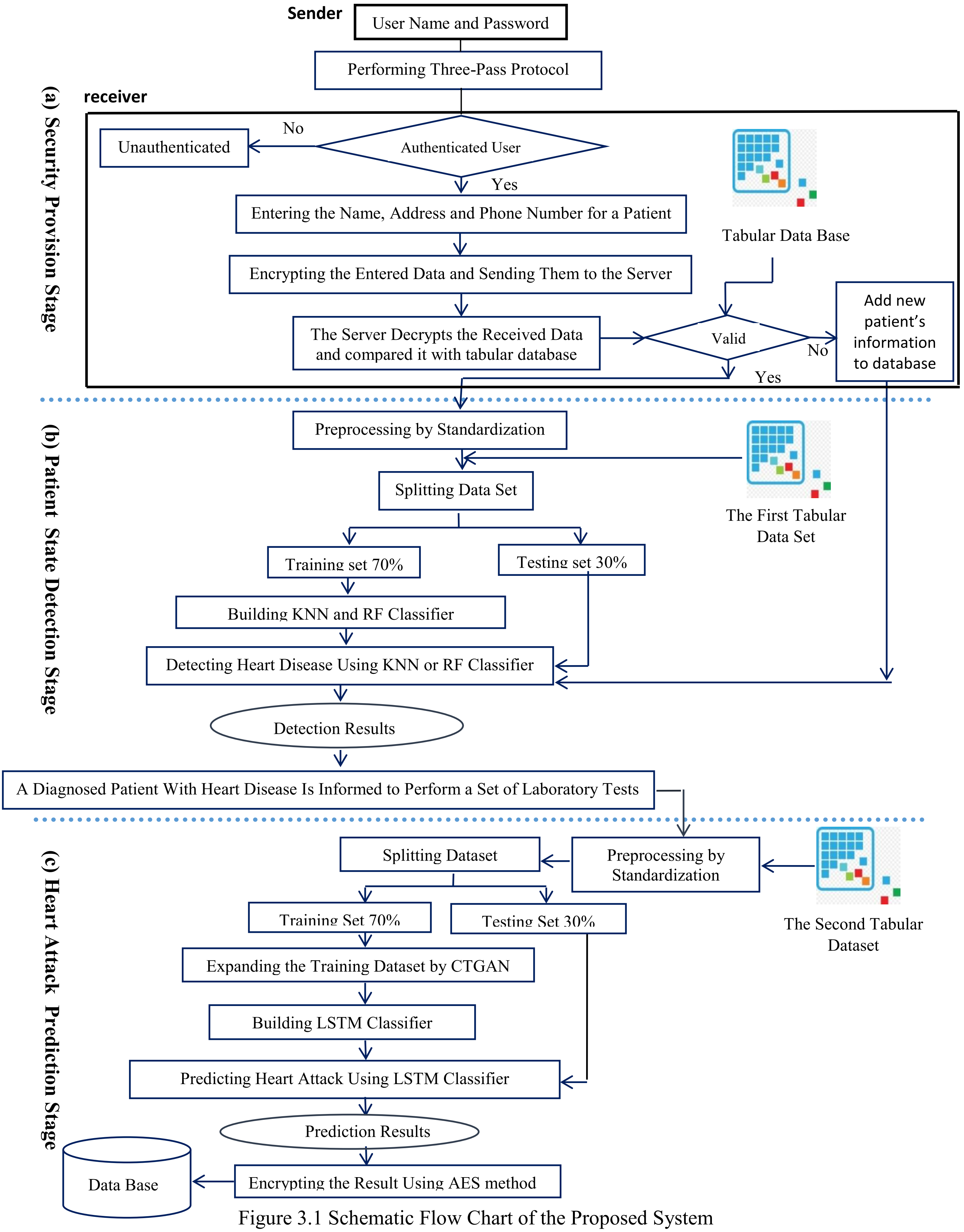

Figure 3.1 Schematic Flow Chart of the Proposed System

The above diagram will be explained in more detail, each stage separately and according to the Figures (3.2),(3.3),(3.4).

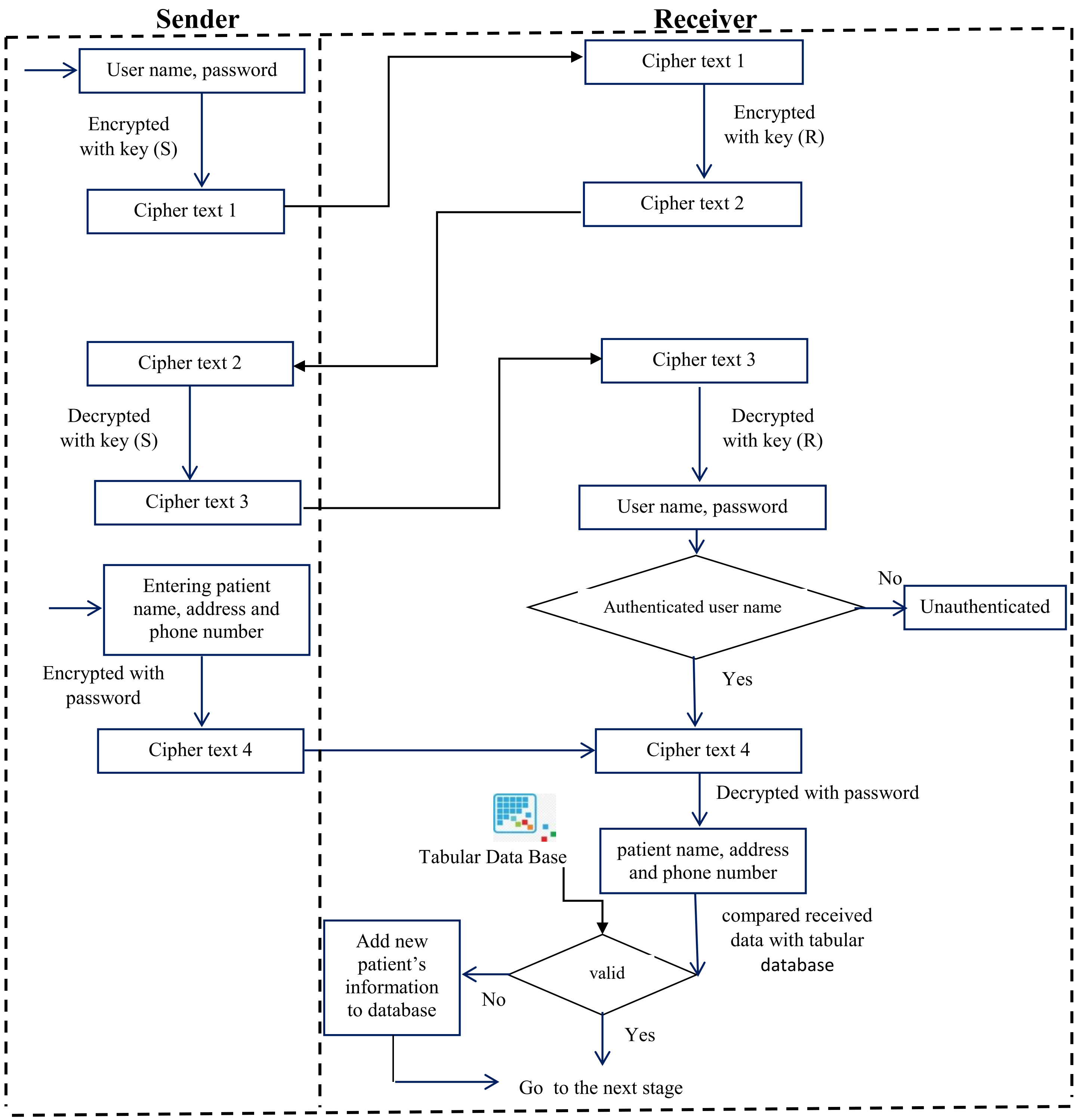

Figure 3.2: Schematic Flow Chart of the Security Provision Stage

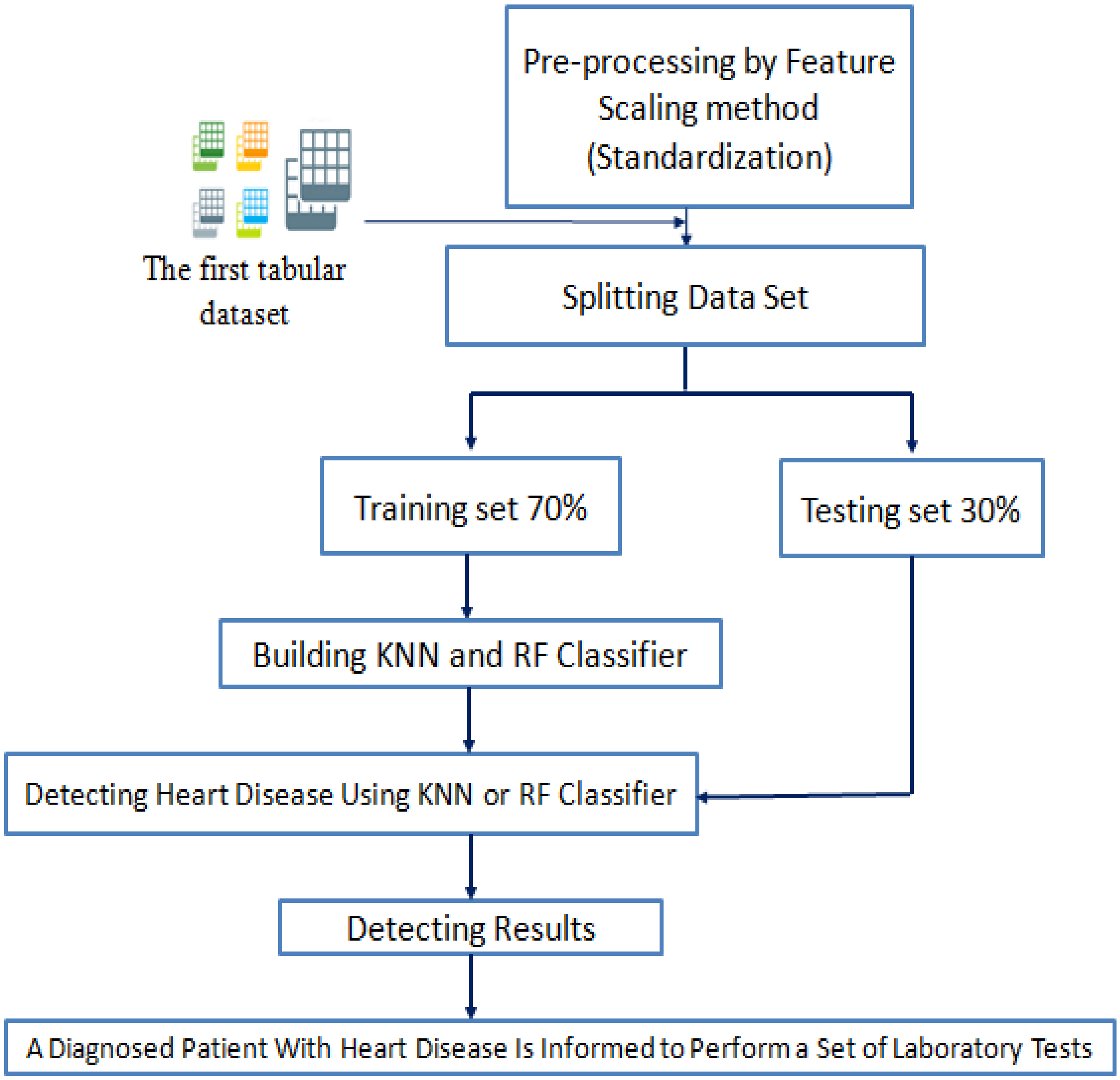

Figure 3.3: Schematic Flow Chart of the Patient State Detection Stage

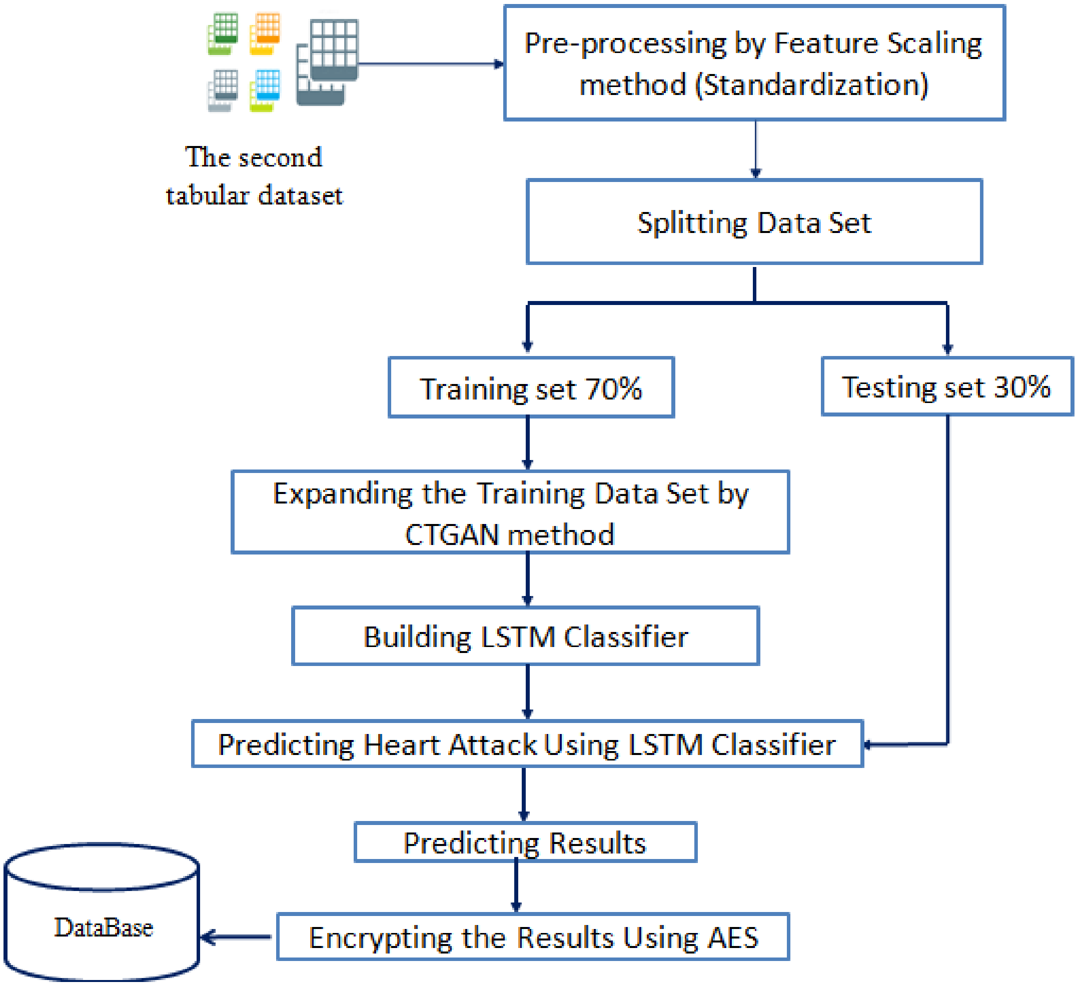

Figure 3.4: Schematic Flow Chart of the Heart Attack Prediction Stage

3.2.1 Security Provision Stage

As previously stated, this stage aims at maintaining the privacy of the data about a patient. To attain that, we securely encrypt the patient's data using the AES encryption technology. Though, distributing the encryption key, which must be safely interchanged between the interested parties, is a drawback of the AES method. By implementing the TP protocol, any user can swap the encryption key without disclosing any sensitive information. This protocol is also used to verify the user authentication by asking this user to share his/her key and name. Essentially, when TPP ends, the server will inspect the received message, which includes the user's name and the AES key, if they match the saved ones, the user is regarded as an honest participant, else, he/she is not. To be more precise, the TPP specification for this stage will be modified as follows:

$$1. A \rightarrow B: \{AES\ key, User's\ name\}_{K(A)}$$

$$2. B \rightarrow A: \{AES\ key, User's\ name\}_{K(A)}\}_{K(B)}$$

$$3. A \rightarrow B: \{AES\ key, User's\ name\}_{K(B)}$$

Figure 3.2 shows a sketch of this protocol which is made up of three passes:

1) **First Pass**: in this pass, the sender with an identity *S* sends a message, that consists of the user's name and the AES key, to the receiver who has the identity *R*. The message should be encrypted under the sender's secret key. Note that the encryption and decryption processes are achieved by Xoring the message with the key.
2) **Second Pass**: upon receiving the message related to the first pass by the receiver, it will encrypted again by the receiver under its secret key. Following that. The receiver sends the encrypted message back to the sender.
3) **Third Pass**: once the sender receives the double encrypted message from the receiver, it decrypts this message using its secret key, and the obtained decrypted will be send to the receiver. Lastly, the receiver also performs the decryption process under its secret key. As a result of the decryption, the receiver now owns the user's name and the AES key.

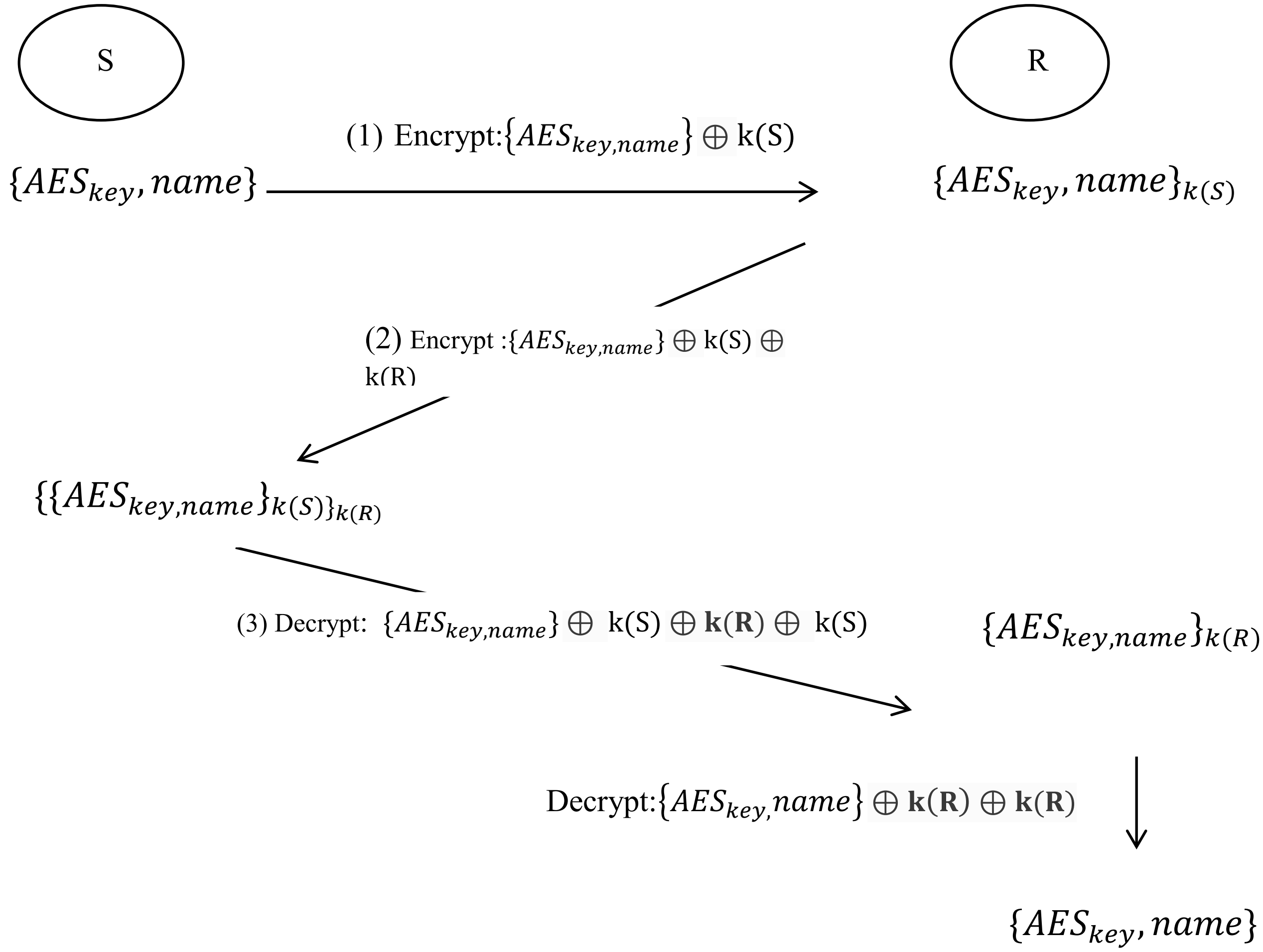

Figure 3.5 The Conceptual Scheme of Three Pass Protocol

When the first authentication is successfully achieved using the TP protocol, the second authentication can be started by sending an encrypted message using the distributed key to the server. This message includes the patient's name, address, and phone number. The server will use the distributed key to decrypt the message once it has been received. Only when the decryption procedure succeeds, the next stage starts.

3.2.2 Patient State Detection Stage:

The main goal of this stage is to select an algorithm that will be effective in accurately identifying whether or not a patient has heart disease. Four machine learning algorithms were implemented, namely Random Forest (RF) and K-Nearest Neighbor (KNN). Age, gender and the number of major vessels are all clinical signs (as shown in Table (3.1)) that are tested by these algorithms.

Table 3.1 First Dataset Attributes Names

Name	Explanation	Measurements
age	the patient's age	years (45-90)
sex	the patient's gender	female = 0, male = 1
cp	kind of chest discomfort	0: standard angina. 1: atypical an angina 2: pain is not angular 3: without symptomatic
trestbps	tresting heart rate	mm Hg
chol	cholesterol	mg/dl
fbs	fasting glucose levels > 120 mg/dl	true = 1, false = 0
restecg	electrocardiograms obtained when unmoving	0: natural 1: exhibiting a distorted ST-T wave 2: showing potential or actual left ventricular hypertrophy according to Estes' standards.
thalach	reached highest heart rate	220 - age of patient
exang	angina brought on by exercise	Yes = 1, no = 0
oldpeak	compared to rest, exercise causes ST depression.	male = 0.2, female = 0.15
slope	the peak workout angle of the ST segment	0: upward slope 1: level 2: downward slope
ca	amount of large vessels	(0 - 3) fluoroscopy-induced color
thal	thalassemia is a blood condition.	0 = normal 1 = fixed defect 2 = defect that is fixable plus the label

Name	Explanation	Measurements
condition	Diagnosis of the patient's condition infected or not infected	0 = no disease, 1 = disease

There are the following sub-stages in the patient state detection stage:

3.2.2.1 Pre-processing Stage:

It's possible that the values of the data in this set were entered by hand, compiled from a variety of sources, and then made available by various government agencies because our system uses an accessible dataset. This means that in order to use these values for classification, preprocessing is required. Feature scaling is the method we use in our proposed system, despite the fact that there are other ways to perform the pre-processing action. It was used to reduce time and complexity. There are several methods for feature scaling, but in this thesis the standardization method was used. It is a technique for distributing the data's independent features in a set range in an equal manner. It regulates extremely variable magnitudes or values. If one of the feature scaling methods is not used and the values of the data set are dissimilar, the performance of the algorithm used will be affected, as it will tend more towards large values, giving them priority in training, and ignoring small values, and thus it will greatly affect the accuracy of the results.

3.2.2.2 Splitting Data Stage:

The testing set and the training set, which are both used to help machine learning techniques produce accurate results, which is used to evaluate the system's performance, are created from the accessible dataset that our system uses. In essence, A training set makes up 70% of the data, while a test set makes up 30%.

3.2.2.3 Detection Stage:

Given that proposed system has potential to influence directly. both the diagnosis of health status and human life, the KNN and RF are two well-known machine learning classifiers that are compared to determine which is superior based regarding the evaluation metrics.

3.2.3 Heart Attack Prediction Stage:

This stage's main objective is to forecast the severity of a heart patient's stroke that results in his death. To accomplish this goal, the LSTM algorithm will be used, and after this algorithm has been applied to the data set that includes a number of important indicators like age, anemia, and other (as shown in Table (3.2)).

Table 3.2 Second Dataset Attributes Names

Name	Explanation	Measurements
age	age of the patient	years(45-90)
anemia	reduction in red blood cells or hemoglobin	Yes = 1, no = 0
creatinine-phosphokinase	blood CPK enzyme concentration	mcg/L
diabetes	the presence or absence of diabetes in the patient	Yes = 1, no = 0
ejection_fraction	what proportion of blood leaves the heart after each contraction	Percentage
high_blood_pressure	regardless of whether the patient has hypertension	Yes = 1, no = 0
Platelets	A blood platelet's presence	kilo platelets/mL
serum_creatinine	concentration of blood creatinine	mg/dL
serum_sodium	the blood sodium level	meq/L
sex	gender of the patient	female = 0, male = 1
smoking	the patient smokes or not	Yes = 1, no = 0

Name	Explanation	Measurements
time	Observation period	days
Death_event	whether the patient passed away while under follow-up care	Yes = 1, no = 0

This stage consists of these sub-stages as follows:

3.2.3.1 Pre-Processing Stage and Splitting Data Stage:

In these stages, the initial processing of the data set will be performed, and then the data will be separated in the same way as mentioned in the previous stage.

3.2.3.2 Data Augmentation Stage:

To artificially increase the amount of data, a group of techniques referred to as "data augmentation" are used to produce new data points from already-existing data. This includes using deep learning models to add new data points or make a few minor changes to the existing data. Data augmentation enhances machine learning models' performance and output by generating fresh, new examples for training datasets. If the dataset is adequate in size, a machine learning model performs better and more accurately. It can take a lot of time and money to gather and label data for machine learning models. Companies can lower these operational costs by transforming datasets using techniques for data augmentation. In our work, we used the CTGAN method, which is a group of artificial data generators for single table data that use deep learning. These generators have the capacity to learn from actual data and generate synthetic data with a high level of fidelity. We will go over a condensed explanation of GAN to help explain the work of CTGAN.

The category of deep learning generator networks that GANs belong to is a supervised learning problem in which we want to add to the set of real data we already have by using a generator. GANs learn to create samples, which is fundamentally different from learning distributions. The discriminator and the generator are two neural networks that make up GANs. While the discriminator tries to distinguish between real and fake data, the generator creates fresh data. The training objectives of the two networks are antagonistic. In contrast to the generator, which seeks to deceive the discriminator, the discriminator seeks to maximize classification accuracy (i.e., correctly identifying which tabular dataset comes from the generator). After learning, the generator ought to be able to create tabular datasets that closely resemble the actual dataset [62].

After using the CTGAN method, we will measure its performance using NewRowSynthesis metric, where the ratio was **1.0**, which is the best performance of the metric. This metric measures whether each row in the synthetic data is novel, or whether it exactly matches an original row in the real data.

3.2.3.3 Prediction Stage:

To find out how often heart patients suffer from cardiac arrest, we will use LSTM algorithm.

After that, the efficiency of the LSTM performance was tested using evaluation scales(Accuracy, Precision, Recall, $F_{measure}$)

After implementing the LSTM algorithm on the patient data set, and for the purpose of maintaining the confidentiality of this data and protecting it from infiltration or hacking, will be encrypted it using the AES algorithm and store it in the database.

Chapter Four

Experimental Results and Evaluation

4.1 Introduction

This chapter presents and discusses the experimental results which are obtained from applying our proposed system. Essentially, the experimental findings for each stage belonging to our system are described in depth. These include: results of conducting the TP protocol together with the AES algorithm, results of both K-Nearest Neighbor (KNN), Random Forest (RF), Naïve Bayes (NB), and Support Vector Machine (SVM) classifiers, and finally the results of LSTM predictor. This chapter also evaluates these attained results by using a set of performance metrics.

4.2 Hardware and Software Specifications

The proposed system is implemented by using HP laptop with 16 GB RAM, Intel Core i7-1165G7 running at 2.80 GHz, and Windows 10 Pro-64-bit operating system. Programmatically, this projects carried out through utilizing Python 3.9.12 language.

4.3 Results of the Proposed System

In this section, the utilized dataset and the experimental results are manifested with respect to the three stages that constitutes our system as well.

4.3.1 Dataset Setting

The database of the first stage is collected manually from random patients. Fundamentally, this database contains personal information, such as: patient's name, phone number, and residential address, see Table (4.1).

Table 4.1 The Employed Database in the Security Provision Stage

Attribute	Description	Measurement
Name	patient name	Alphabet (A…Z)
Address	The patient's residential address	Name of city
Phone Number	The patient's mobile number	Number

The second stage, on the other hand, makes use of an online dataset[1] called **Kaggle** which contains different features, such as: age, sex, chestpain type, etc. These features represent the main health conditions that rise the risk of heart disease. In fact, the dataset in this stage include 14 features and 303 patients. This dataset is divided into two sets: 70% training set and 30% testing set.

Whereas the dataset[2] of the third stage includes 13 features and 299 patients, such that these features comprise: age, anemia, platelets, serum_creatinine, serum_sodium, sex, etc.

In order to pre-processing the dataset, we apply the Feature Scaling (Standardization method) before conducting the classification and prediction algorithms. The main reason for that is remove the inconsistent data values by scaling them in to a range [0:1].

(1) The dataset is available online at (https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci)

(2) The dataset is available online at (https://www.kaggle.com/code/nayansakhiya/heart-fail-analysis-and-quick-prediction/input)

4.3.2 Security Provision Stage

The Three-Pass protocol is suggested deal with key distribution issue of the AES algorithm. In addition, by utilizing this protocol and AES encryption, any attempt to access the system will be subjected to double authentication. The implementation of the TPP can be seen in the following example:

$plain\ text = ali, AES_{key}$

$$binary\ value\ of\ the\ plain\ text = \ 0110000101101100 \ldots\ldots\ldots\ldots$$

$key(Sender) = key(A) = 10011111$

$key(Reciever) = key(B) = 01011010$

Accordingly, the steps of the TPP are as follows:

1. $0110000101101100 \ldots\ldots\ldots\ldots \quad [plain\ text]$

   $10011111 \quad [key(A)]$

   $11111110 \ldots. [the\ firts\ cipher\ text\ produced\ by$
   $XORing\ the\ plaintex\ with\ the\ key\ and$
   $then\ A\ sends\ the\ result\ to\ the\ B]$

2. $11111110 \ldots. \quad [the\ firts\ cipher]$

   $01011010 \quad [key(B)]$

   $10100100 \ldots\ldots\ldots. \quad [the\ second\ cipher\ text\ produced\ by$
   $XORing\ the\ first\ cipher\ text\ with\ the$
   $key\ then\ B\ sends\ the\ result\ to\ the\ A]$

3. $10100100 \ldots\ldots\ldots. \quad [the\ second\ cipher\ text]$

   $10011111 \quad [key(A)]$

   $00111011 \ldots\ldots \quad [the\ first\ decryption\ produced\ by$
   $XORing\ the\ second\ cipher\ text\ with$
   $the\ key, then\ A\ sends\ the\ result\ to\ the\ B$

4. $00111011\ldots\ldots$ $[the\ first\ decryption]$
01011010 $[key(B)]$
$01100001\ldots\ldots.[the\ second\ decryption\ which\ will\ be\ converted\ to\ the\ binary\ to\ obtain\ the\ message\ and\ the\ key\ that\ must\ be\ processed\ by\ AES$

From the above steps, it can be noticed that the sender and the receiver securely sends the message and the key without sharing secret keys between them in advance.

In order to **evaluate** the TPP, the MITM attack have launched by assuming that the Xor encryption method is known by the attacker as shown in the steps below:

1.1 $A \rightarrow I(B)$: $\{name, AES_{key}\}_{k_A}$

2.1 $I(B) \rightarrow A$: $\{name, AES_{key}\}_{k_A}$

2.2 $A \rightarrow I(B)$: $name, AES_{key}$

1.2 $I(B) \rightarrow A$: $randtext$

1.3 $A \rightarrow I(B)$: $\{randtext\}_{k_A}$

2.3 $I(B) \rightarrow A$: $\{randtext\}_{k_A}$

Note that the number on the left side represents the session's number, while the number on the right side represents the actual step number of the protocol, *randtext* acts as a random textual message generated by the attacker, the first session is between *A* participant (the sender) and the attacker (the receiver) that impersonates *B* participant, and the second session, on the other hand, is between the attacker (the sender) that impersonates *B* participant and *A* participant (the receiver).

However, the MITM attack can be prevented if *A* checks that it does not receive a message encrypted with its key.

With respect to the evaluation of the AES method, the following metrics are undertaken:

a) **Dictionary Attack**: this attack is launched depending on two strategies (as mentioned in Chapter 2). The dictionary attack with its two strategies does not succeed, as keys containing numbers have randomly generated, characters, and special symbols, and these keys are not found in the English dictionary.

b) **Avalanche effect metric** is used to analyze the AES method. Fundamentally, four trials have been conducted with the aim at indicating that changing only one bit leads to a good degree of diffusion of the output cipher text. In other words, the cryptanalysis attack is hard to achieve. Table 4.2 shows the experimental results for these trials, which are conducted by using the AES method to encrypt the original plain text that is equal to:

"**a**lib7625e224dc0f0ec91ad28c1ee67b1eb96d1a5459533c5c950f44aae1e32f2da36d1a5459533c5c950f44aae1e32f2da31eb96d1a5459533c5c950f44aae1e32f2da36d1a5459533c5c950f44aae1e32f2da3".

From this table, we can see that the AES method has an average avalanche effect that equals to 49.88.

Table 4.2 Avalanche Effect Test Results

The Changed Message	Flipped Bits Number	Avalanche Effect
dlib7625e224dc0f0ec91a…………	622	50.36
elib7625e224dc0f0ec91a…………	617	49.96
flib7625e224dc0f0ec91a…………	601	48.66
glib7625e224dc0f0ec91a………….	624	50.53
Average	616	49.88

c) **Key space**: As we used a key length of 128 bits, the AES method has an effective key space of 2^{128}=3.402823669209385e+38. As a result, this increases the time required by the brute-force attack.

4.3.3 Patient State Detection Stage

Before starting the diagnosis stage, we will explain the stage of dividing the patient data set used in this stage, as the data set contains 14 features and **303** rows, and it split into two groups: the test set of data by 30%, and the training set by 70%, so the training data set has rows that total approximately is **212** rows whereas there are **91** rows in the test data set.

In the second stage, classifiers like Random Forest and K-Nearest Neighbor are used to determine whether or not a person has the cardiac disease. The identical patient data set is applied individually to each classifier. Several measurement parameters, including accuracy, precision, sensitivity, and F1 score, are used to evaluate the classifier's performance. According to the results, the Random Forest approach outperforms the metrics of the K-Nearest Neighbor algorithm utilized.

Table 4.3 The Effectiveness of the Random Forest Classification Method

Name of Method	Amount of trees	accuracy	precision	sensitivity	F1_score
Random Forest (RF) number of trees = 1,2,3,5	1	0.98	0.98	0.98	0.98
	2	0.96	0.97	0.96	0.96
	3	**0.99**	0.99	0.99	0.99
	5	**0.99**	0.99	0.99	0.99

Table (4.3) displays the Random Forest categorization technique's performance results when there are 1, 2, 3, and 5 trees. This table shows that a tree population of 3,5 results in the greatest value for various parameters. The accuracy values are shown in Figure (4.1) where its appearance is the accuracy and the n-estimators.

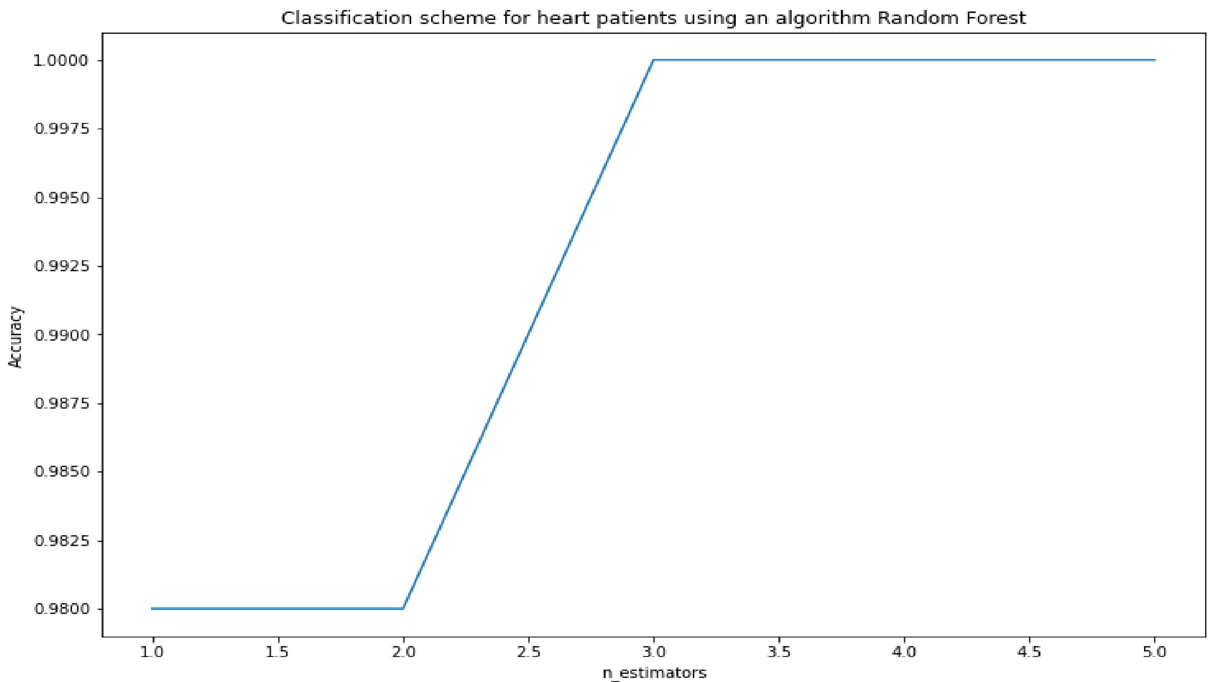

Figure 4.1 The Accuracy of Random Forest

Figure (4.1) displays a series of RF algorithm experiments that were conducted and their results by varying the number of trees, or n_estimators, value, with the greatest outcomes being reached when (n_estimators) = 3 and 5.

Also, the confusion matrix for accuracy value and n_estimators values are shown in figures ((4.2),(4.3),(4.4), and (4.5)).

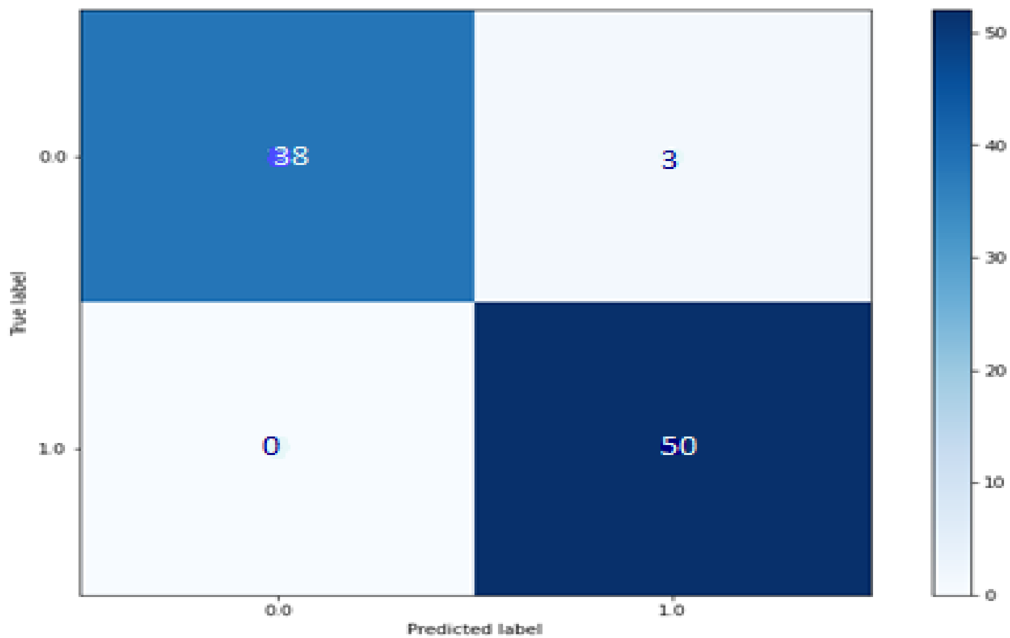

Figure 4.2 The Confusion Matrix of Random Forest where n_estimator = 1

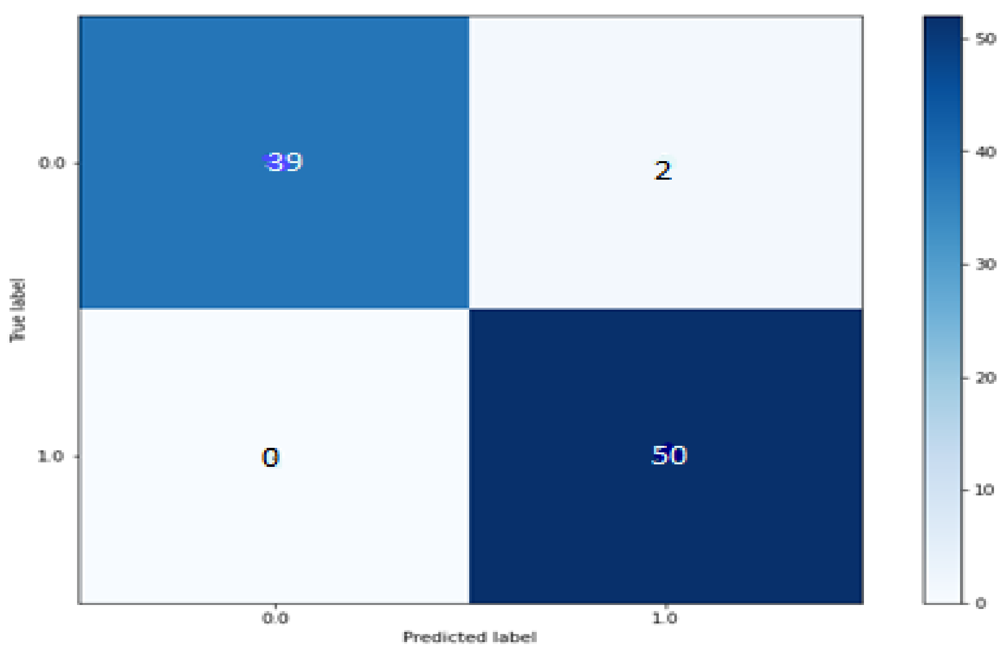

Figure 4.3 The Confusion Matrix of Random Forest where n_estimator = 2

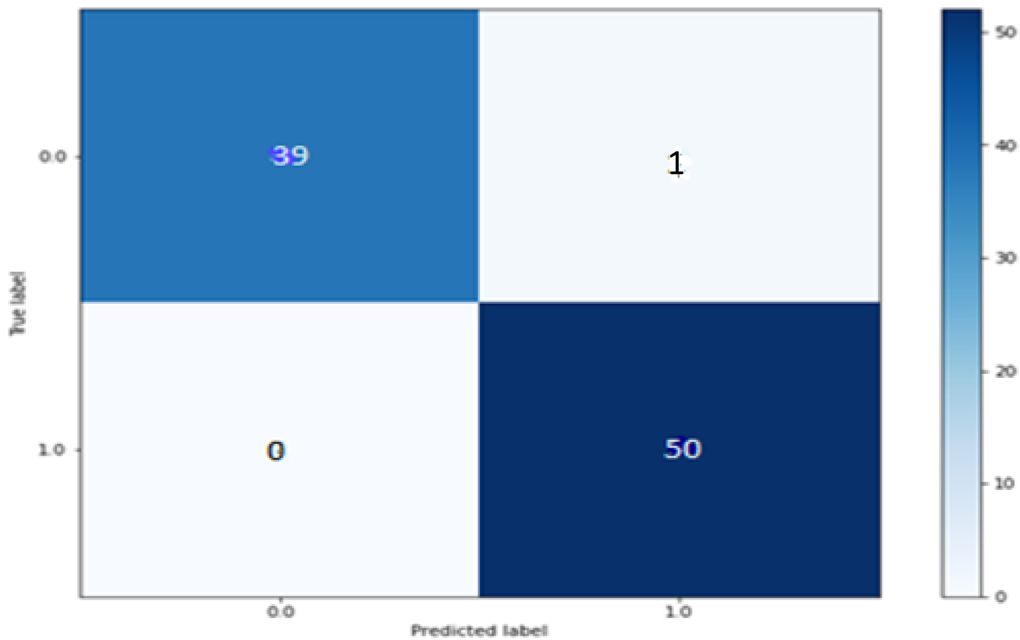

Figure 4.4 The Confusion Matrix of Random Forest where n_estimator = 3

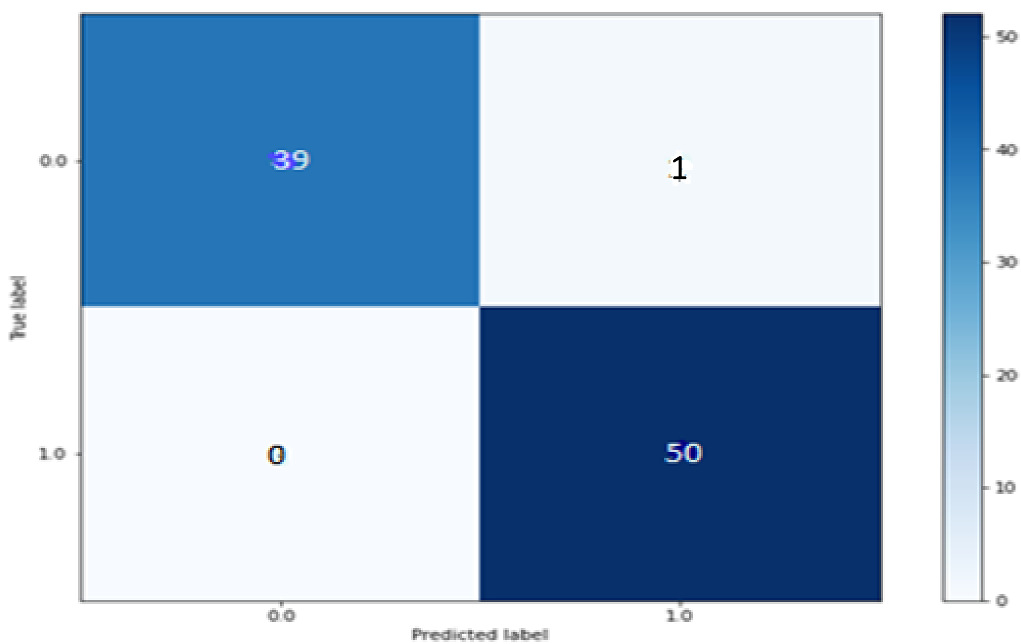

Figure 4.5 The Confusion Matrix of Random Forest where n_estimator = 5

The K-Nearest Neighbor classifier results are shown in Table (4.4) when the number of neighbors(K) is set to 3,5,7 or 9. The value of K was found to produce the best results for the chosen metrics when it is 5.

Table 4.4 The K-Nearest Neighbor Classification Technique's Performance Results

Name of Method	Number of neighbors	Accuracy (%)	Precision	Sensitivity	F1_score
K-Nearest Neighbor (KNN) number of neighbors = 3,5,7,9	3	0.94	0.94	0.94	0.94
	5	**0.96**	0.96	0.96	0.96
	7	0.95	0.95	0.95	0.95
	9	0.95	0.95	0.95	0.95

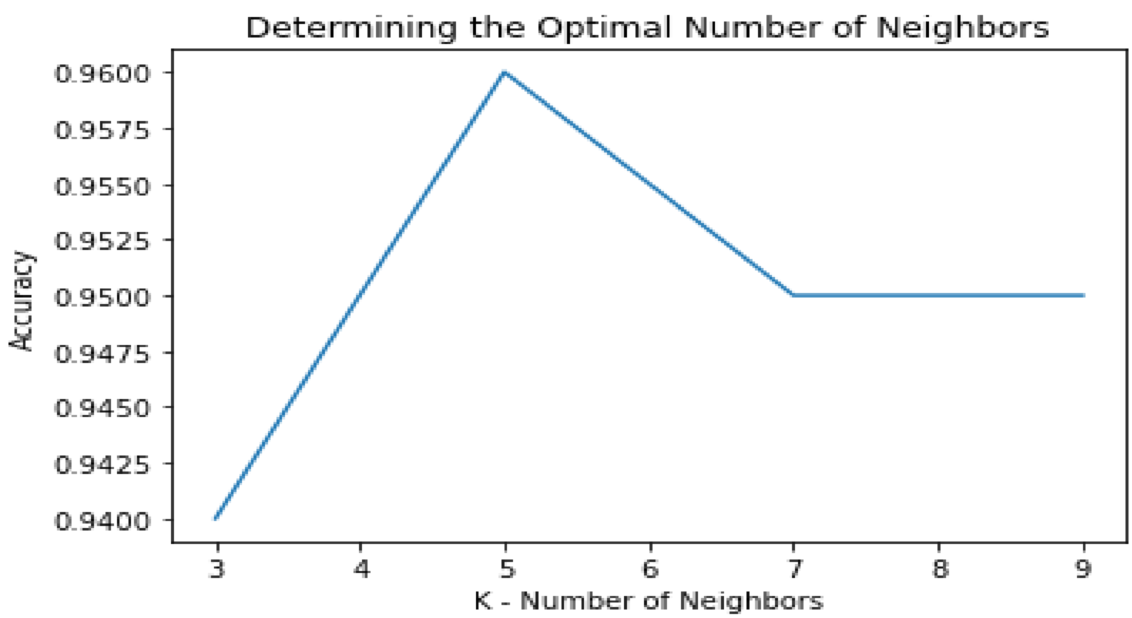

Figure 4.6 The Accuracy of K-Nearest Neighbor

Figure (4.6) illustrates a series of tests that were done to demonstrate how different K values can impact how well the KNN algorithm performs. Typically, K=5 yields the best outcomes. Also, the confusion matrix for accuracy value and K values are show in figures (4.7), (4.8), (4.9), and (4.10).

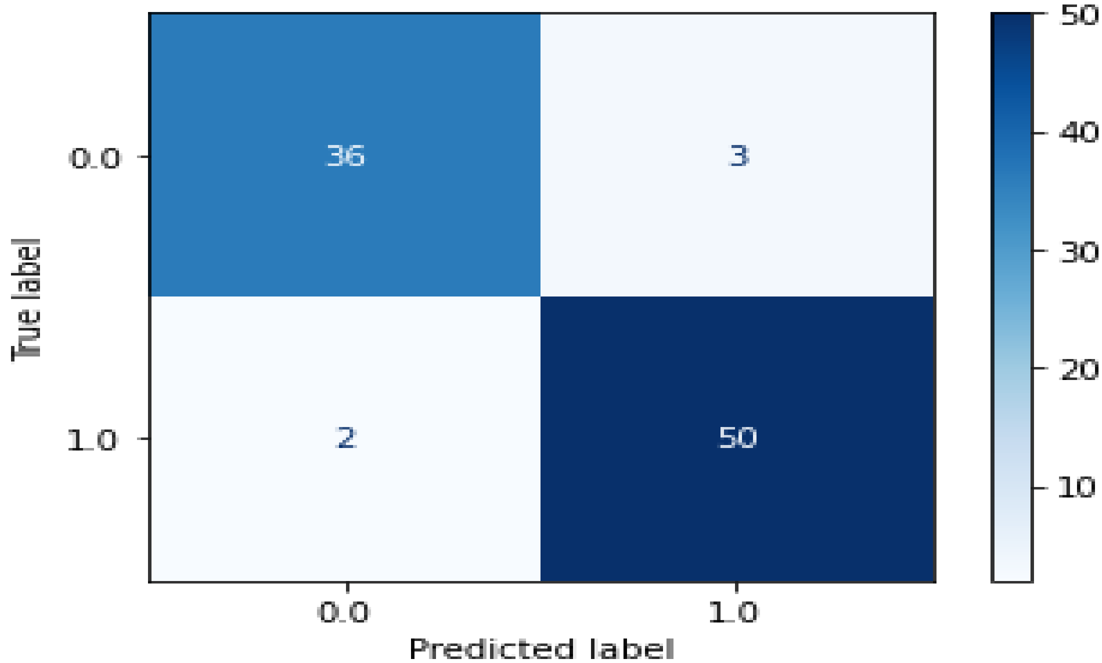

Figure 4.7 The Confusion Matrix of KNN where K = 3

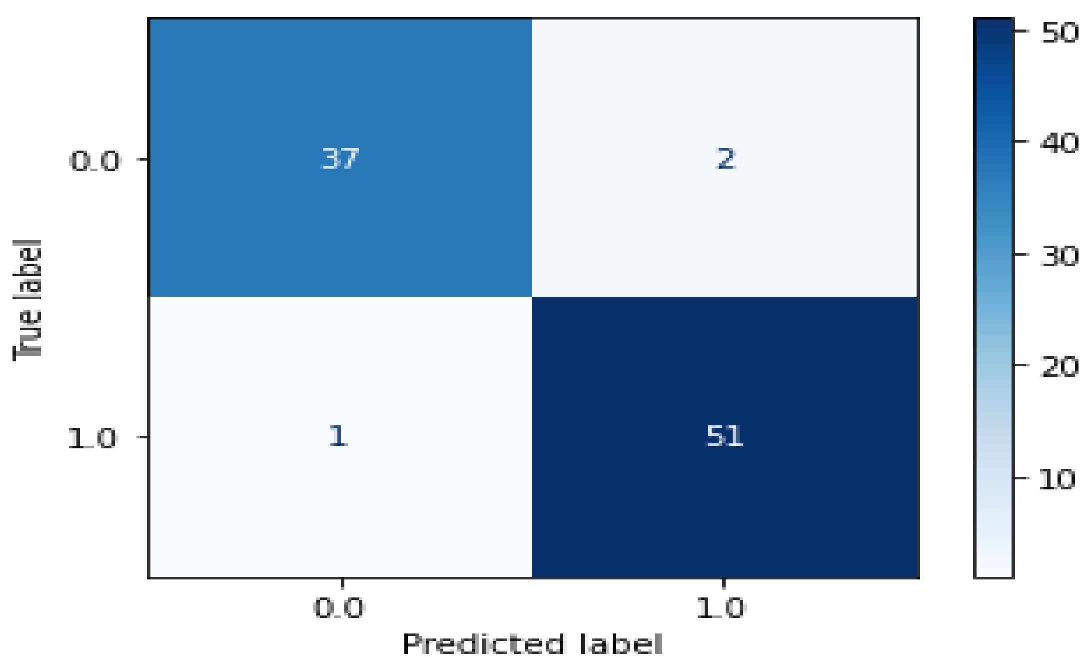

Figure 4.8 The Confusion Matrix of KNN where K = 5

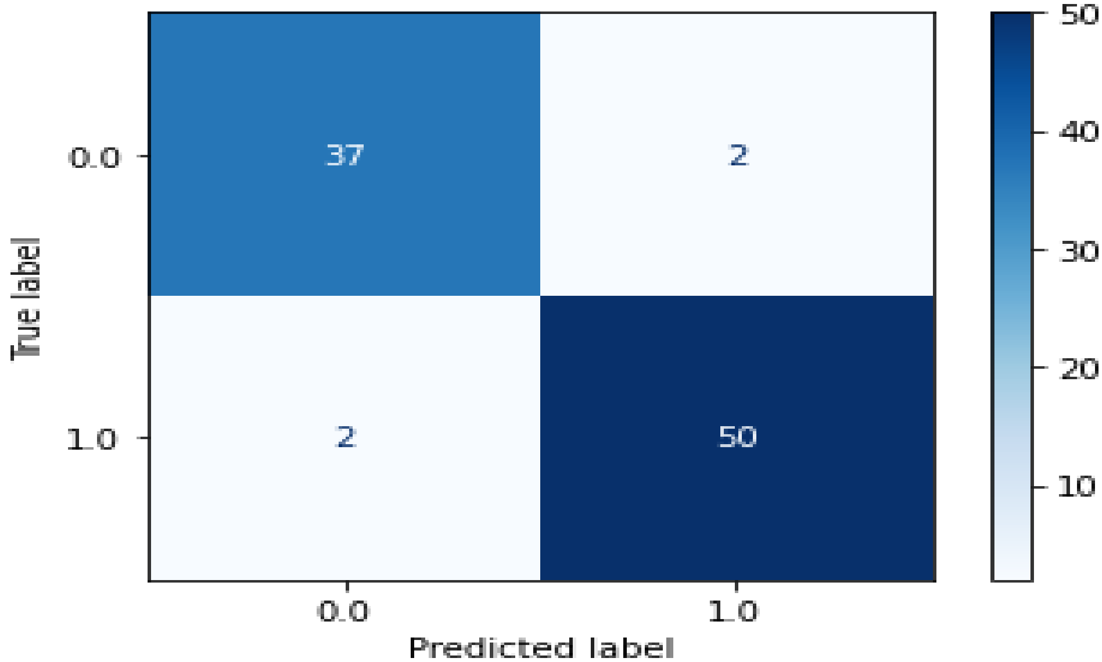

Figure 4.9 The Confusion Matrix of KNN where K = 7

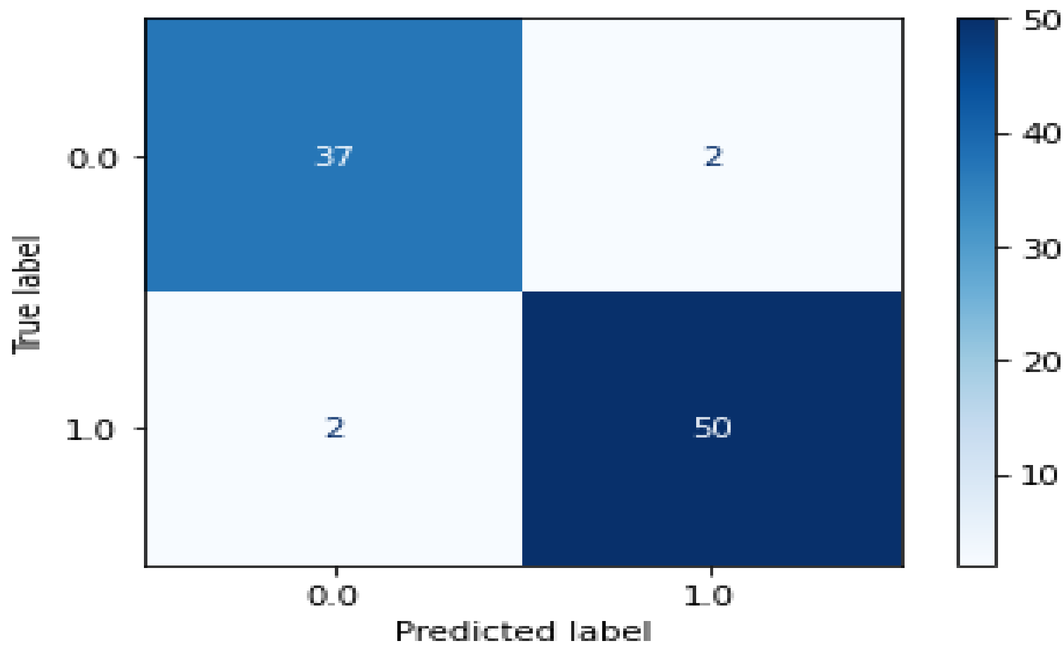

Figure 4.10 The Confusion Matrix of KNN where K = 9

The SVM classifier results are shown in Table (4.5), while Figure (4.11) illustrates the accuracy and Figure (4.12) illustrates the confusion matrix for this algorithm.

Table 4.5 The Support Vector Machine Classification Technique's Performance Results

Name of Method	Accuracy (%)	Precision	Sensitivity	F1_score
Support Vector Machine	0.93	0.90	0.92	0.93

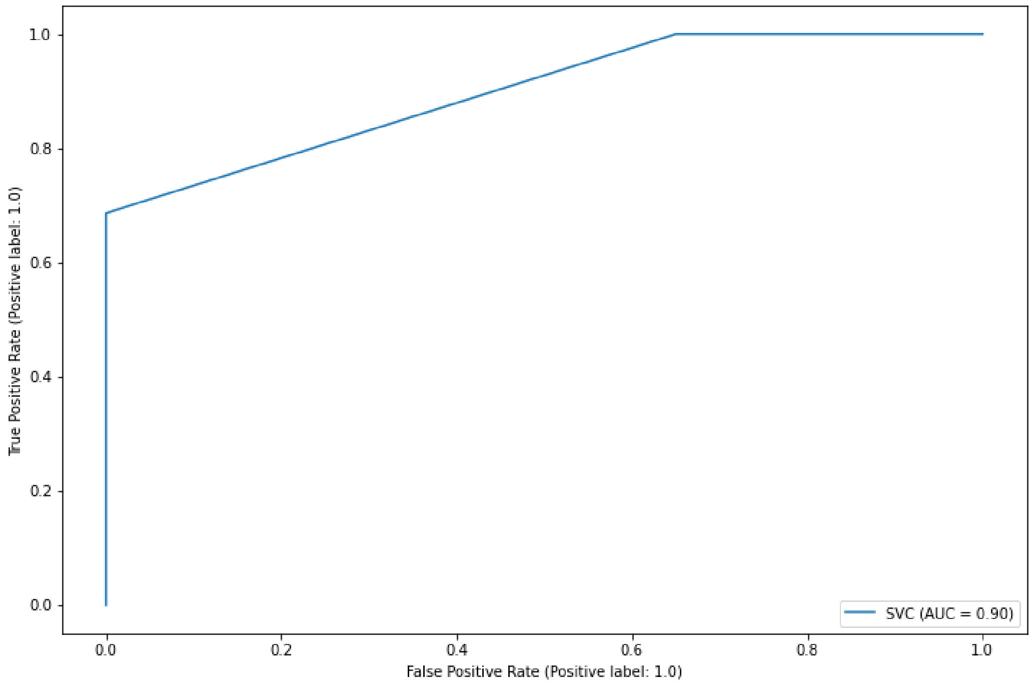

Figure 4.11: The Accuracy of Support Vector Machine

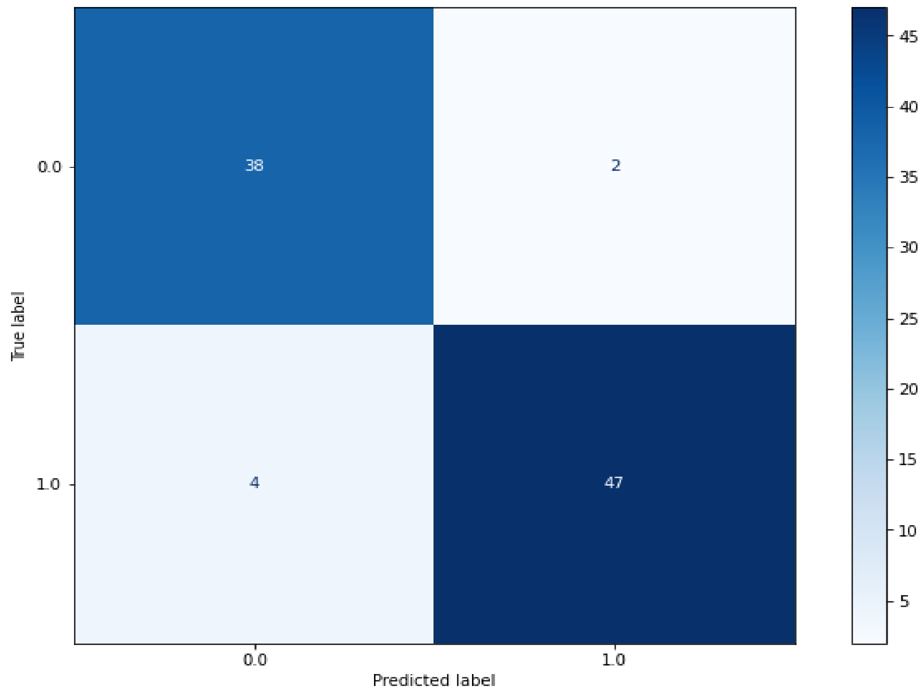

Figure 4.12 The Confusion Matrix of Support Vector Machine

The NB classifier results are shown in Table (4.6), while Figure (4.13) illustrates the accuracy and Figure (4.14) illustrates the confusion matrix for this algorithm.

Table 4.6 The Naïve Bayes Machine Classification Technique's Performance Results

Name of Method	Accuracy (%)	Precision	Sensitivity	F1_score
Naïve Bayes	0.81	0.86	0.73	0.75

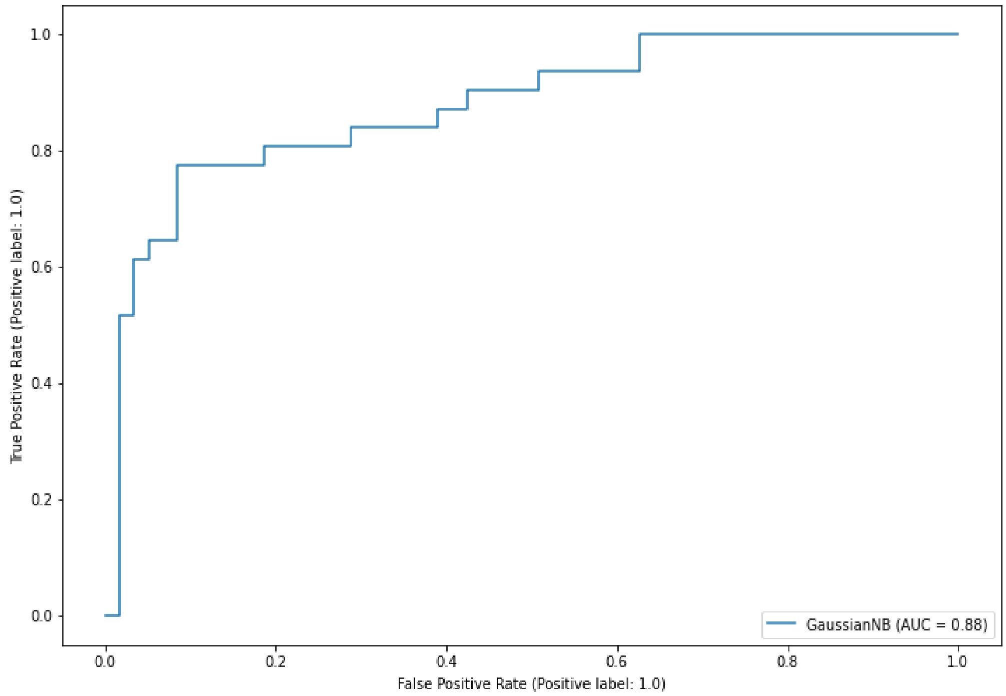

Figure 4.13: The Accuracy of Naïve Bayes

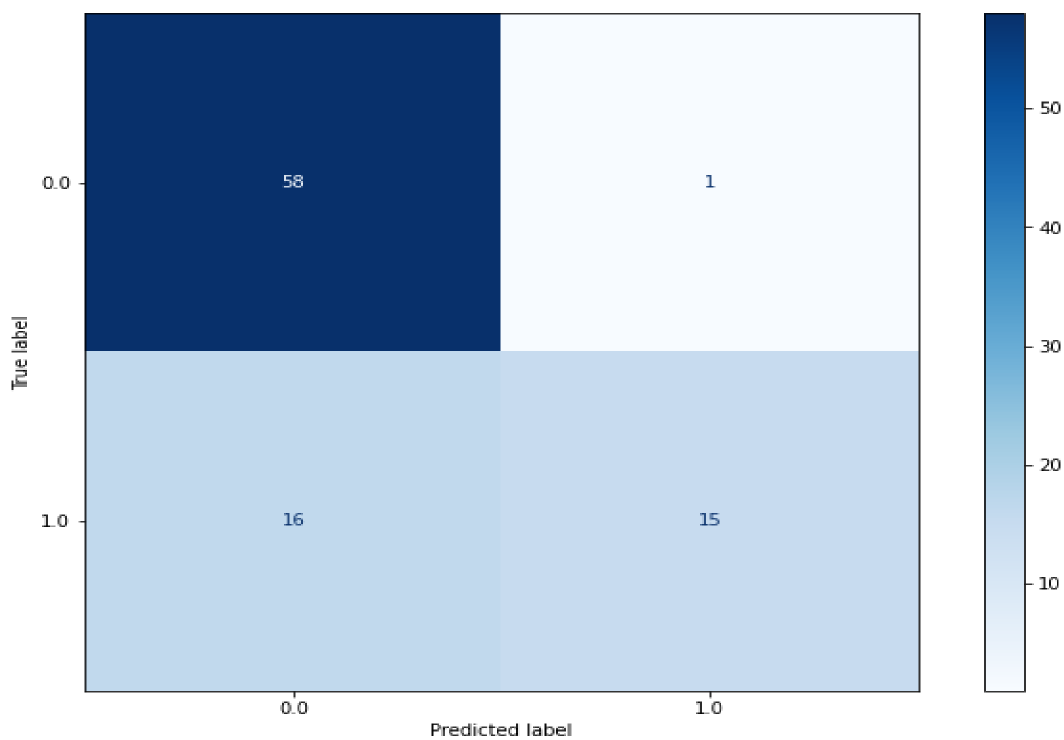

Figure 4.14 The Confusion Matrix of Naïve Bayes

based on the adopted measures identified in this study, Figure (4.15) shown compares Knn, RF, SVM and NB. Where this graph shows how the RF algorithm outperforms on other algorithms.

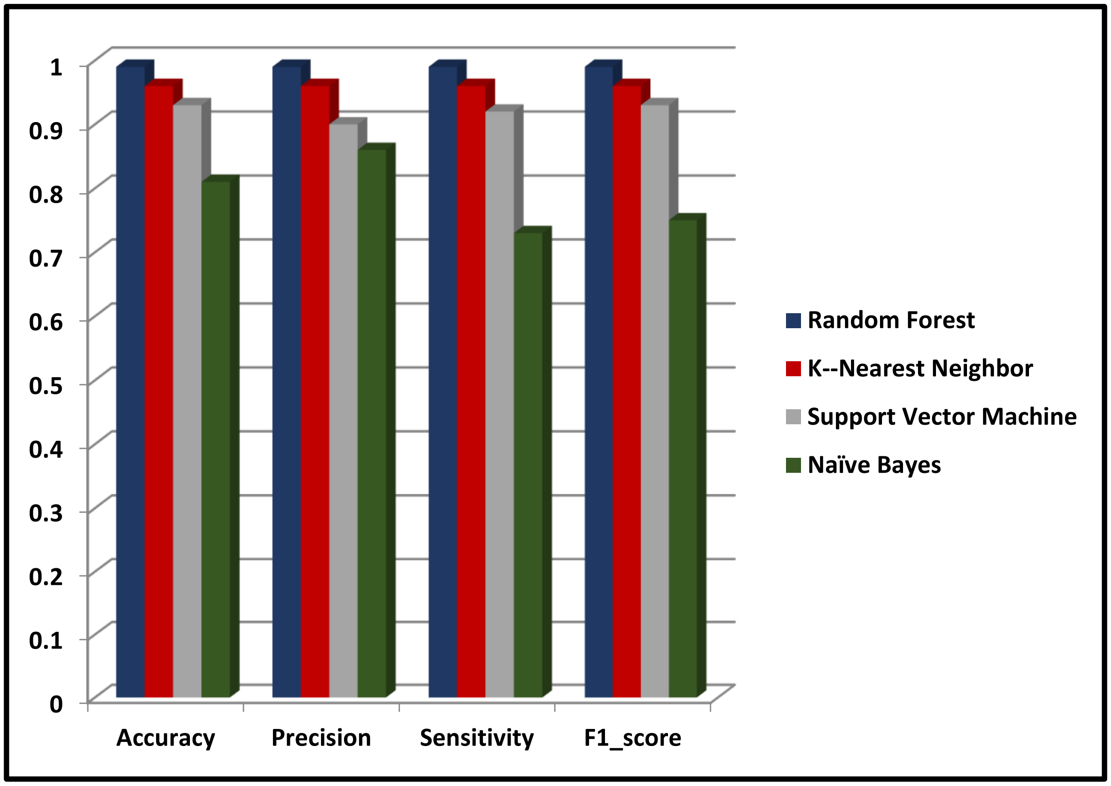

Figure 4.15 Performance Evaluation of RF, KNN, SVM and NB algorithms

4.3.4 Heart Attack Prediction Stage

Before starting with the prediction stage, we will explain the segmentation stage of the cardiac patient data set used in this stage, as the data set contains 13 features and **299** rows, and it is split into two groups: the training set of data by 70% and the test set of data by 30%, so the training data set has rows that total approximately is **220** rows whereas there are **79** rows in the test data set.

In the third stage, the implementation LSTM algorithm on the patient data set will be in two sub-stage:

First: Implementation LSTM algorithm on the original data set before augmenting it as shown in the results of following table:

Table 4.7 The Performance Results of the Long Short-Term Memory Prediction Technique Before Data Augmentation

Accuracy (%)	Precision	Sensitivity	F1_score
0.65	0	0	0

In this work, there is a problem which is the small size of the patient data set used where after implementing the LSTM algorithm on a patient data set, the results show that the accuracy is not good as shown in Table (4.7). Figure (4.16) shows the unreasonable and unacceptable training set and test set curves for the patient data set after implementing the LSTM algorithm on it.

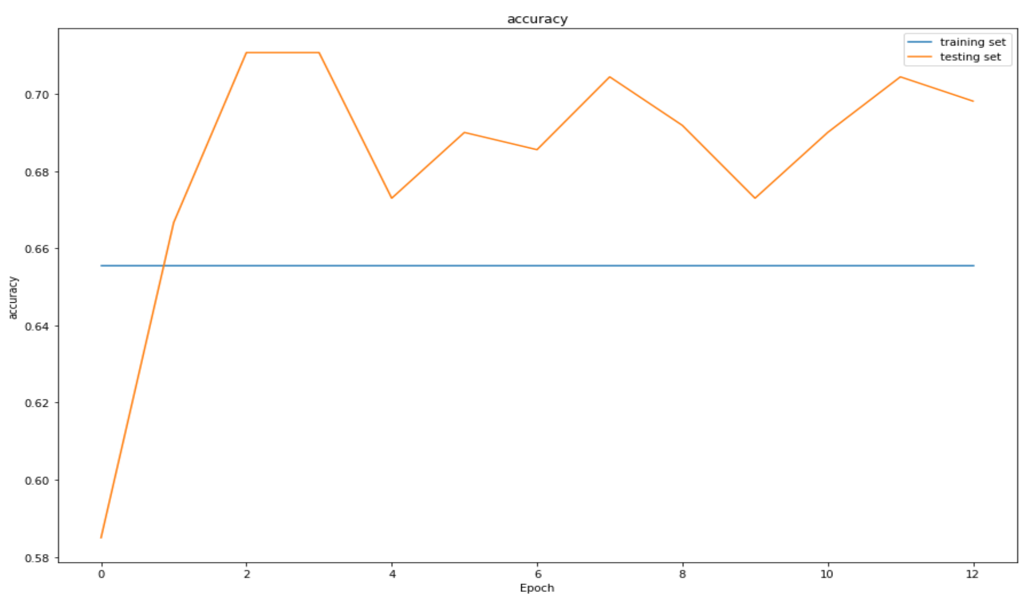

Figure 4.16 The Accuracy of LSTM

So, increasing the size of this data using the CTGAN method was suggested. Where the results after implementing this technique showed an increase in the number of rows in the original data set from **299** rows to **5000**. This increase is very useful and the resulting data set can be used in several applications.

The **NewRowSynthesis** measure, which is one of the well-known performance measures, where obtained a **ratio of 1.0**, which is the highest performance rate for the scale used.

Second: After augmentation of the cardiac patient's data set and using the CTGAN method, the number of rows increased from **299** to **5000** rows, and the training data set received a 70% split, where as the test data set received a 30% split, so the number of rows for the training data set was approximately **3773**, while it was For the test data set it is **1227**. After that, The LSTM algorithm was implemented on this data set, and the results were as shown in the table below.

Table 4. 8 The Performance Results of the Long Short-Term Memory Prediction Technique After Data Augmentation

Accuracy (%)	Precision	Sensitivity	F1_score
0.99%	0.99%	0.99%	0.99%

After executing the LSTM algorithm on the data set after augmentation Table (4.8) shows that the accuracy has increased significantly compared to the accuracy mentioned in Table (4.7).

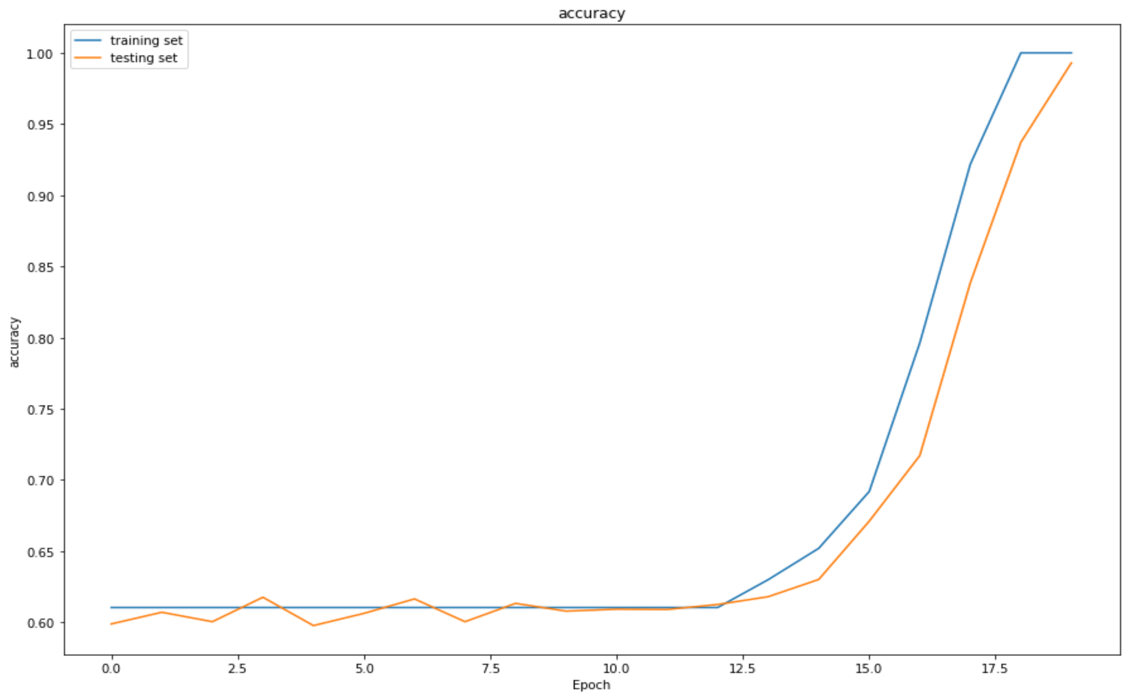

Figure 4.17 The Accuracy of LSTM

Figure (4.17) shows that the curves of the training and test groups increase gradually and regularly after implementing the LSTM algorithm on the patient data set after augmentation.

Also, the confusion matrix for the accuracy value of LSTM algorithm is shown in Figure (4.18).

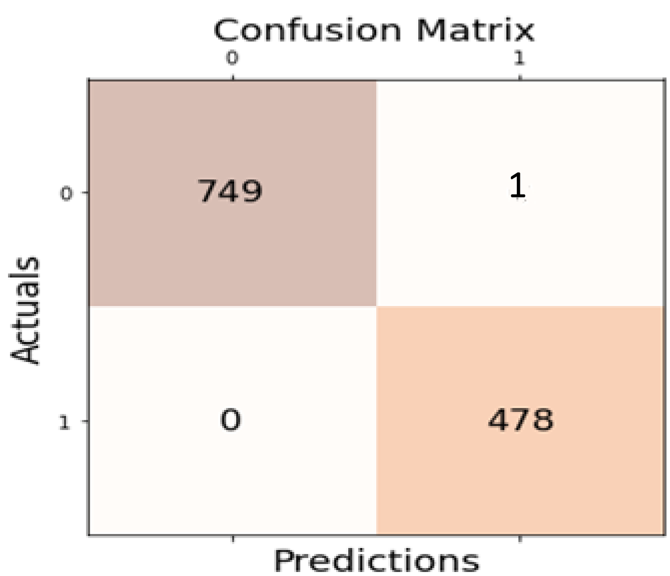

Figure 4.18 The Confusion Matrix of LSTM

Figure (4.19) depicts a comparison between the Long Short Term Memory algorithm before data augmentation and after it based on the measures used and established in this work.

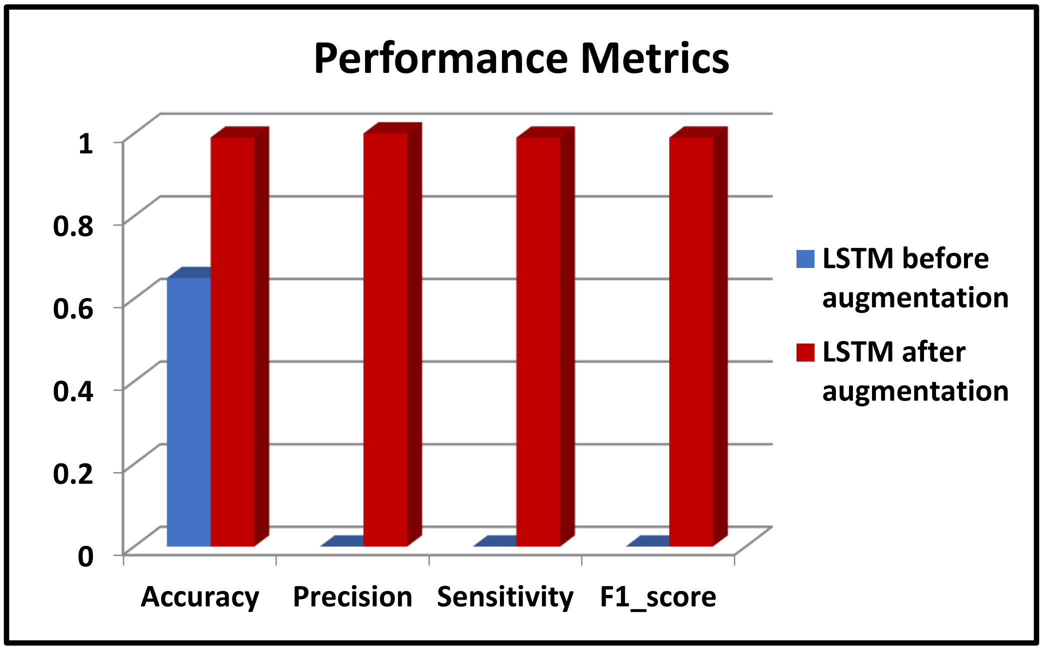

Figure 4.19 The Performance Comparison Between LSTM Before and After Data Augmentation

Table 4.9 Comparison Table with Some Related Work

References	Year	Dataset	Technique(s)	Accuracy
[26]	2021	www.kaggle.com	CTGAN, TGAN methods	49.8%
[30]	2023	www.kaggle.com	XGBoost	87%
Our envisioned work	2023	www.kaggle.com	Random Forest, KNN, AES encryption, LSTM, and CTGAN	KNN = 96% RF = 99% LSTM = 99%

The findings of the current research are contrasted with those of various relevant publications in Table (4.9). This table includes five columns: the citation's number and the year it was published, the data set this reference utilized, the technique this reference used, and the accuracy this reference attained.

Figure (4.20) depicts the performance evaluation of our work's outcomes against those of various relevant efforts. This graph includes the technique utilized, the number of references, and the accuracy of each technique.

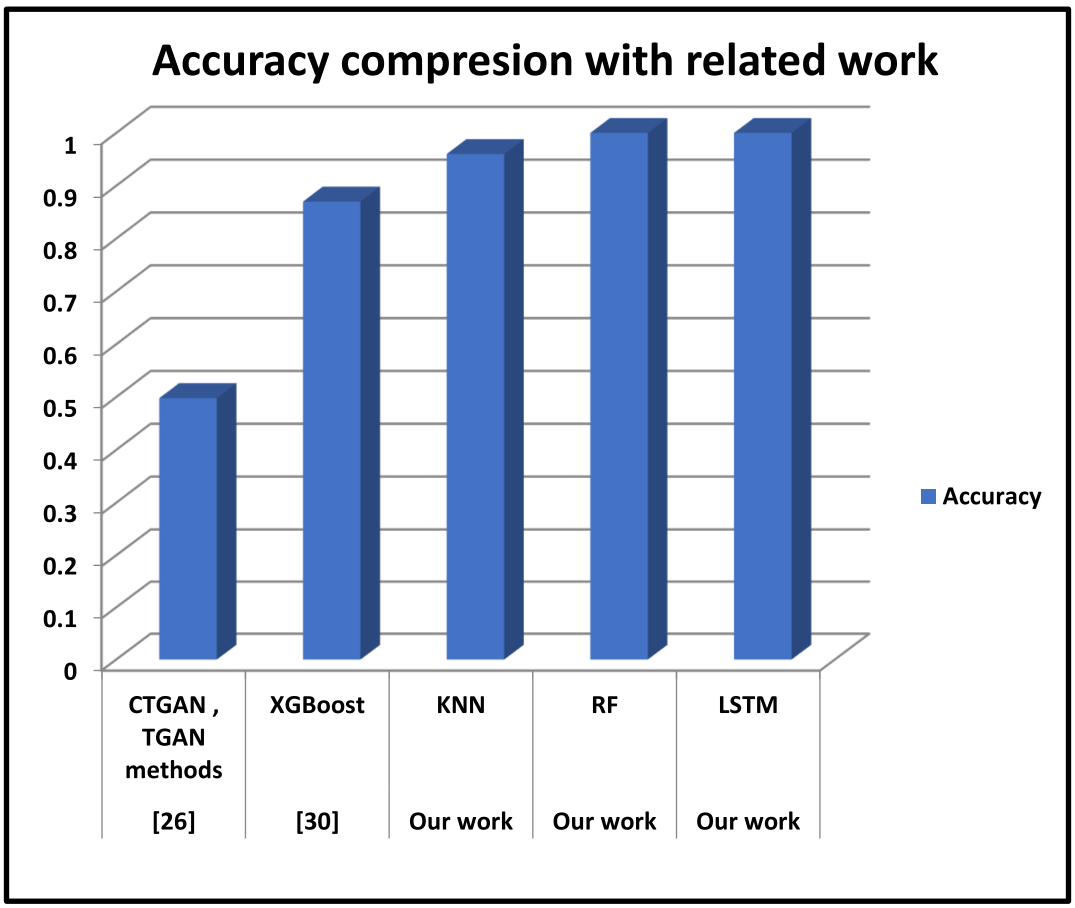

Figure 4.20 The Performance Comparison Proposed Work with Some Related Work

Chapter Five

Conclusions and Future Works

5.1 Conclusions

The current thesis has outlined the following:

1- The MITM attack has been successfully launched against TPP. However, adding the verification action to the honest participant, that is represented by not receiving the same sent messages, can prohibit this attack.

2- The Random Forest algorithm was adopted and gave results that were superior to those of the K-Nearest Neighbor algorithm, where RF accuracy was 99% and KNN accuracy was 96%, when these algorithms were implemented on the first dataset found in the following link: (https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci) by presenting two algorithms and choosing the best one. However, more than one parameter was looked at in the two procedures to enhance performance and choose the best parameter.

3- When applying the LSTM algorithm to the second data set found in the following link: (https://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-data), the accuracy of the algorithm was very low, equal to 65%, precision is zero. So, in order to improve Prediction accuracy of this algorithm The boosting was performed on the previously mentioned data set using the CTGAN method, after that the algorithm was implemented again on the boosted data set and we obtained a very high accuracy equal to 99% and a high accuracy equal to 99% as well, and this is the most important thing in the medical field.

5.2 Suggestions for Future Works

During the research work of this thesis, a number of possible future directions are located. These directions are:

1- Conducting other cryptographic protocols, such as vera crypt protocol.
2- Performing the authentication by scanning the fingerprint, encrypting, and sending it to the authentication service, where it is compared against the user's original schema.
3- This thesis can be updated in the future by using the heart disease dataset to include more attributes and make it more interactive for the users.
4- After training the algorithms the same system can be used on the dataset for other diseases.
5- This model can be used with data sensors to monitor a person's vital signs, and this is one of the applications of IOT.

References

References

[1] G. M. C. Rosano *et al.*, "*Impact analysis of heart failure across European countries: an ESC-HFA position paper*," ESC Heart Failure, Italy, p. 12, 2022.

[2] M. da Veiga, A., Astakhova, L. V., Botha, A., & Herselman, "*Defining organisational information security culture-perspectives from academia and industry*," Computer & Security 92, South Afracia, p. 50, 2020.

[3] The daily swig/cybersecurity news and views, 14/3/2022, "*Data breach at US heart disease treatment center impacts 287,000 individuals*", [Online].Available: https://portswigger.net/daily-swig/data-breach-at-us-heart-disease-treatment-center-impacts-287-000-individuals

[4] L. dwight Smith, "*Cryptography; The Science of Secret Writing*", Courier Corporation, USA, p. 164, 2021.

[5] R. Imam, Q. M. Areeb, A. Alturki, and F. Anwer, "*Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status*," IEEE Access, India, p. 28, 2021.

[6] H. M. Mohammad and A. A. Abdullah, "*Enhancement process of AES: a lightweight cryptography algorithm-AES for constrained devices*," Telkomnika (Telecommunication Comput. Electron. Control., Iraq, p. 10, 2022.

[7] Khasanah, P. T. Nguyen, G. Gunawan, and R. Rahim, "*Three-pass protocol scheme on vigenere cipher to avoid key Distribution*", Journal of Critical Reviews, Indonesia, p. 4, 2020.

[8] D. Nababan and R. Rahim, "*Security Analysis Combination Secret Sharing Protocol and Three-Pass Protocol*," Journal of Physics: Conference Series, Indonesia, p. 9, 2019.

[9] M. Vaduganathan, G. A. Mensah, J. V. Turco, V. Fuster, and G. A. Roth, "*The Global Burden of Cardiovascular Diseases and Risk: A Compass for Future Health*", E L S EVIER, USA, p. 11, 2022.

[10] world Health Organization, 1/1/2023, "*Cardiovascular diseases*", [Online].Available:https://www.who.int/health-topics/cardiovascular-diseases

[11] Y. Yan, J. W. Zhang, G. Y. Zang, and J. Pu, "*The primary use of artificial intelligence in cardiovascular diseases: What kind of potential role does artificial intelligence play in future medicine?*", Journal of Geriatric Cardiology , China, p. 7, 2019.

[12] N. Dutta, U. Subramaniam, and S. Padmanaban, "*Mathematical models of classification algorithm of Machine learning*", International Meeting on Advanced Technologies in Energy, Saudi Arabia, p. 2, 2019.

[13] V. V Aishwarya Mujumdar, "*Diabetes Prediction using Machine Learning Algorithms*", Elsevier B.V., India, p. 8, 2019.

[14] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "*Ensemble deep learning: A review,*" Engineering Applications of Artificial Intelligence, India, p. 47, 2022.

[15] Z. Z. J. W. Yuwei Liu, Yuqiang Cheng, "*Multi-information Fusion Fault Diagnosis Based on KNN and Improved Evidence Theory*." Journal of Vibration Engineering & Technologies, China, p. 13, 2021.

[16] L. Wang, X. Wu, H. Chen, and T. Zeng, "*Prediction of impermeability of the concrete structure based on random forest and support vector machine*," IOP Conference Series: Earth and Environmental Science, China, p. 9, 2020.

[17] R. V. K.Kalaivani, N. Uma Maheswari, "*Heart Disease Diagnosis Using Optimized Features OF Hybridized Alcsoga Algorithm And LSTM Classifier,*", Research Square, India, p. 16, 2022.

[18] K. V. Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, "*Modeling Tabular Data using Conditional GAN*", 33rd Conference on Neural Information Processing Systems, Canada, p. 11, 2019.

[19] C. Zhang, L. Zhu, C. Xu, and R. Lu, "*PPDP: An efficient and privacy-preserving disease prediction scheme in cloud-based e-Healthcare system*", Elsevier B.V., china, p. 10, 2018.

[20] A. N. Repaka, S. D. Ravikanti, and R. G. Franklin, "*Design and implementing heart disease prediction using naives Bayesian*", Proceedings of the Third International Conference on Trends in Electronics and Informatics IEEE, India, p. 6, 2019.

[21] X. Yang, R. Lu, J. Shao, X. Tang, and H. Yang, "*An efficient and privacy-preserving disease risk prediction scheme for E-healthcare*", IEEE, Canada, p. 14, 2019.

[22] L. J. and Y. Z. Kuang Junwei, Hangzhou Yang, "*Dynamic prediction of cardiovascular disease using improved LSTM*", International Journal of Crowd Science, China, p. 12, 2019.

[23] S. R. D. Debjani Panda, Ratula Ray, Azian Azamimi Abdullah, "*Predictive Systems: Role of Feature Selection in Prediction of Heart Disease*", Journal of Physics: Conference Series PAPER, India, p. 7, 2019.

[24] M. B. Abdulaziz Albahr , Marwan Albahar and Mohammed Thanoon, "*Computational Learning Model for Prediction of Heart Disease Using Machine Learning Based on a New Regularizer*", Hindawi, Saudi Arabia, p. 10, 2021.

[25] A. M. · M. I. · Z. M. · A. I. · M. N. · T. Nazir and M. Masood, "*Prediction of Heart Disease Using Deep Convolutional Neural Networks*", springer, Pakistan, p. 15, 2021.

[26] O. L. Min Jong Cheon, Dong Hee Lee, Ji woong Park, Hye Jin Choi, Jun Seuck Lee, "*Ctgan Vs Tgan? Which One Is More Suitable For Generating Synthetic Eeg Data*", Journal of Theoretical and Applied Information Technology, South Korea, p. 15, 2021.

[27] S. S. U. Ch. Anwar ul Hassan, Jawaid Iqbal, Rizwana Irfan, Saddam Hussain, Abeer D. Algarni, Syed Sabir Hussain Bukhari, Nazik Alturki, "*Effectively Predicting the Presence of Coronary Heart Disease Using Machine Learning Classifiers*", Sensors, Pakistan, p. 19, 2022.

[28] D. S. and P. C. Umarani Nagavelli, "*Machine Learning Technology-Based Heart Disease Detection Models*", Hindawi, India, p. 9, 2022.

[29] PanelHuru H. A and M.-S. K. , Muhammad Tufail b , Ui-Jun Baek a , Jee-Tae Park a, "*A novel blockchain-enabled heart disease prediction mechanism using machine learning*," Korea, *Comput. Electr. Eng.*, p. 10, 2022.

[30] C. M. Bhatt and T. G. and P. L. M. , Parth Patel, "*Effective Heart Disease Prediction Using Machine Learning Techniques*", Algorithms, Italy, p. 10, 2023.

[31] J. Yang and T. Johansson, "*An overview of cryptographic primitives for possible use in 5G and beyond*," Science China Information Sciences., Sweden, p. 22, 2020.

[32] R. Elhabob, Y. Zhao, I. Sella, and H. Xiong, "*An efficient certificateless public key cryptography with authorized equality test in IIoT*," Springer Berlin Heidelberg, China, p. 21, 2020.

[33] S. Kumar, M. S. Gaur, P. Sagar Sharma, and D. Munjal, "*A Novel Approach of Symmetric Key Cryptography*," IEEE, India, p. 7, 2021.

[34] C. Tezcan, "*Optimization of Advanced Encryption Standard on Graphics Processing Units*," IEEE Access, Turky, p. 12, 2021.

[35] B. Langenberg, H. Pham, and R. Steinwandt, "*Reducing the Cost of Implementing the Advanced Encryption Standard as a Quantum Circuit*," IEEE Transactions on Quantum Engineering, USA, p. 12, 2020.

[36] Fathurrahmad and Ester, "*Development and Implementation of The Rijndael Algorithm and Base-64 Advanced Encryption Standard (AES) for Website Data Security*," International Journal of Computer Applications., Indnoesia p. 5, 2020.

[37] Research Gate, 1/9/2018, "*A General Process of AES Algorithm*" [Online]. Available: https://www.researchgate.net/figure/Modified-AES Algorithm_fig1_326837307

[38] K. Bhargavan *et al.*, "*DY*: A modular symbolic verification framework for executable cryptographic protocol code*," IEEE European Symposium on Security and Privacy, Germany, p. 21, 2021.

[39] Panda mediacenter, 21/2/2022, "*The Main Concept of Man-In-The-Middle Attack*", [Online]. Available: https://www.pandasecurity.com/en/mediacenter/security/man-in-the-middle-attack/

[40] Rublon, 13/3/2023, "*The Main Concept of Dictionary Attack*", [Online]. Available: https://rublon.com/blog/brute-force-dictionary-attack-difference/

[41] P. K. Sahoo and P. Jeripothula, "*Heart Failure Prediction Using Machine Learning Techniques*", SSRN Electronic Journal, India, p. 14, 2021.

[42] H. K. Bhuyan and V. Ravi, "*Analysis of Sub-feature for Classification in Data Mining*", IEEE Transactions on Engineering Management., India, p. 17, 2021.

[43] M. Batta, "*Machine Learning Algorithms - A Review",* International Journal of Science and Research (IJSR), India, p. 7, 2020.

[44] J. Heidari, "*Classifying Material Defects with Convolutional Neural Networks and Image Processing*", UPPSALA University, Sweden, p. 56, 2019.

[45] M. A. Wani, F. A. Bhat, S. Afzal, and A. I. Khan, "*Advances in Deep Learning"*, springer , Poland, p. 159, 2019.

[46] Y. M. Wazery, E. Saber, E. H. Houssein, A. A. Ali, and E. Amer, "*An Efficient Slime Mould Algorithm Combined with K-Nearest Neighbor for Medical Classification Task*", IEEE Access, Egypt, p. 17, 2021.

[47] Mediam, 3/3/2021, "*KNN algorithm structure*" [Online]. Available: https://ai.plainenglish.io/introduction-to-k-nearest-neighbors-knn-algorithm-e8617a448fa8

[48] Informatic, 22/12/2017, "*Steps of KNN algorithm*" [Online]. Available: https://informatic-ar.com/k-nearest-neighbor-algorithm/

[49] F. S. Alotaibi, "*Implementation of machine learning model to predict heart failure disease*", International Journal of Advanced Computer Science and Applications, Saudi Arabia, p. 8, 2019.

[50] Mediam, 23/2/2021, "*RF algorithm structure*" [Online]. Available: https://medium.com/analytics-vidhya/random-forest-classifier-and-its-hyperparameters-8467bec755f6

[51] Simplilearn, 26/2/2023, "*Steps of RF algorithm*" [Online]. Available: https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forestalgorithm#:~:text=Step%201%3A%20Select%20random%20samples,as%20the%20final%20prediction%20result.

[52] R. Geirhos *et al.*, "*Shortcut learning in deep neural networks*", Nature Machine Intelligence, Germany, p. 29, 2021.

[53] P. S. Lu´ıs Santos, Filipe N. Santos, Paulo Moura Oliveira, "*Deep Learning applications in agriculture: a short review*", Advances in Intelligent Systems and Computing, Portugal, p. 13, 2020.

[54] C. Z. Xinting Yang, Song Zhang, Jintao Liu, Qinfeng Gao, Shuanglin Dong, "*Deep learning for smart fish farming: applications, opportunities and challenges*", Reviews in Aquaculture, China, p. 46, 2021.

[55] J. Wang, Y. Zou, P. Lei, R. Simon Sherratt, and L. Wang, "*Research on recurrent neural network based crack opening prediction of concrete dam*", Journal of Internet Technology, China, p. 10, 2020,

[56] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, "*Recurrent neural network wave functions*", PHYSICAL REVIEW RESEARCH 2, Canada, p. 17, 2020.

[57] J. Guo *et al.*, "*Efficient minimum word error rate training of RNN-Transducer for end-to-end speech recognition*", Amazon.com, USA, p. 5, 2020.

[58] Medium, 3/9/2020, "*Deep RNN structure*" [Online]. Available: https://medium.com/swlh/simple-explanation-of-recurrent-neural-network-rnn-1285749cc363

[59] R. Zhao, J. Xue, J. Li, W. Wei, L. He, and Y. Gong, "*On Addressing Practical Challenges for RNN-Transducer*", IEEE, China, p. 8, 2021.

[60] J. C. Kim and K. Chung, "*Prediction model of user physical activity using data characteristics-based long short-term memory recurrent neural networks*", KSII Transactions on Internet and Information Systems. South Korea, p. 18, 2019.

[61] Opengenus, 5/2/2023, " *GAN-based techniques*" [Online]. Available: https://iq.opengenus.org/types-of-gans/

[62] M. Mendikowski and M. Hartwig, "*Creating Customers That Never Existed: Synthesis of E-commerce Data Using CTGAN*," 18th International

Conference on Machine Learning and Data Mining, Germany, p. 14, 2022.

[63] Datacebo, 20/12/2022, "*CTGAN Schema*" [Online]. Available: https://datacebo.com/blog/interpreting-ctgan-progress/

[64] M.-P. Cote, B. Hartman, O. Mercier, J. Meyers, J. Cummings, and E. Harmon, "*Synthesizing Property & Casualty Ratemaking Datasets using Generative Adversarial Networks*", arXiv, USA, p. 19, 2020.

[65] GitHub, 26/9/2022, "*NewRowSynthesis metric*", [Online]. Available: https://docs.sdv.dev/sdmetrics/metrics/metrics-glossary/newrowsynthesis

[66] R. M. Teasdale, "*Evaluative Criteria: An Integrated Model of Domains and Sources*", American Journal of Evaluation, USA, p. 24, 2021.

[67] N. D. Jorstad and J. Landgrave T. Smith, "*Cryptographic algorithm metrics*", 20th National Information Systems Security, Virginia, p. 38, 1997.

[68] H. K. Obayes, F. S. Al-Turaihi, and K. H. Alhussayni, "*Sentiment classification of user's reviews on drugs based on global vectors for word representation and bidirectional long short-term memory recurrent neural network*", Indonesian Journal of Electrical Engineering and Computer Science, Iraq, p. 9, 2021.

[69] X. Wan, "Influence of feature scaling on convergence of gradient iterative algorithm", Journal of Physics: Conf. Series, china, p.6, 2019.

[70] Data Science and machine learning, 29/4/2019, "*Confusion-matrix*" [Online]. Available: https://www.debadityachakravorty.com/images/cmatrix.jpg

[71] M. Edida, N. J. Lakshmi, and N. Jabalia, "*Automated Diagnosis of Diabetes Mellitus Based on Machine Learning*", Springer, India, p. 19, 2021.

الخلاصة

تركز هذه الأطروحة على ثلاث قضايا تتعلق بتشخيص أمراض القلب وأمن البيانات التشخيصية. أولاً، النظر في مشكلة توزيع المفاتيح المرتبطة بأنظمة تشفير المفاتيح المتماثلة. ثانيًا، اختيار تقنية مناسبة للتعلم الآلي قادرة على تشخيص مشاكل القلب بدقة وفي الوقت المناسب. وأخيرًا، توسيع مجموعة بيانات التدريب لتحسين أداء التنبؤ لتقنية التعلم العميق. ونتيجة لذلك، تقوم هذه الأطروحة ببناء نظام تشخيص آلي لأمراض القلب من ثلاث مراحل. تتناول المرحلة الأولى مشكلة المفتاح المتماثل من خلال استخدام بروتوكول Three-Pass لتوزيع المفتاح السري. تم اختيار هذا البروتوكول نظرًا لقدرته على تسهيل النقل الآمن للرسائل بين الأطراف دون الحاجة إلى معرفة مسبقة أو مشاركة المفاتيح. يتم تقييم هذه المرحلة من خلال إطلاق هجمات القاموس وهجمات الرجل في الوسط (MITM). يفشل هجوم القاموس لأنه يتم إنتاج المفاتيح عشوائيًا باستخدام أرقام وأحرف ورموز أخرى غير موجودة في المعجم الإنجليزي. نجح هجوم MITM، ولكن إضافة خطوة التحقق المتمثلة في عدم إرسال نفس الرسائل إلى المشاركين الصادقين يمكن أن يمنعه. في المرحلة الثانية، تم تحليل خوارزميات K-Nearest Neighbor (KNN) وRandom Forest (RF) باستخدام معايير أداء مختلفة لحل المشكلة الثانية. يتم استخدام مجموعة بيانات من العلامات الحيوية لتصنيف المريض على أنه يعاني من مرض القلب في هذه المرحلة. تظهر نتائج مقياس الدقة أن خوارزمية Random Forest (99%) تفوقت على KNN (96%). بينما تتم معالجة مجموعات بيانات التدريب غير الكافية في المرحلة الثالثة لزيادة أداء الذاكرة طويلة المدى (LSTM). في الأساس، يتم تطبيق منهج شبكة الخصومة التوليدية الجدولية المشروطة من أجل توسيع مجموعة بيانات التدريب. تنمو صفوف مجموعة البيانات من 299 إلى 5000. وتكشف النتائج التجريبية أن دقة تنبؤات LSTM تزيد من 65% إلى 99%.

إ

نظام تشخيص آمن لأمراض القلب يعتمد على التعلم الآلي

رسالة مقدمة الى

مجلس كلية العلوم للبنات-جامعة بابل

وهي جزء من متطلبات نيل درجة الماجستير في علوم الحاسبات

من قبل

رندة شاكر عبد الحسين

بإشراف

أ.د. هضاب خالد عبيس

د. فرح محمد حسن الشريفي

2023م 1445هـ