Republic of Iraq
Ministry of Higher Education and Scientific Research
University of Al-Qadisiyah
College of Computer Science and Information Technology

A Proposed Approach for ECG Signal Classification Using One Dimensional CNN

A Thesis
Submitted to the Council of the College of Computer Science and Information Technology at the University of Al-Qadisiyah in Partial Fulfilment of the Requirements for the Degree of Master in Computer Science

by

Sarah Kamil Katfaan

Supervised by:

Assist Prof. Dr. Lamia AbedNoor Mohammed

2021 A.D. **1443 A.H.**

بسم الله الرحمن الرحيم

﴿فَبَدَأَ بِأَوْعِيَتِهِمْ قَبْلَ وِعَاءِ أَخِيهِ ثُمَّ اسْتَخْرَجَهَا مِن وِعَاءِ أَخِيهِ كَذَٰلِكَ
كِدْنَا لِيُوسُفَ مَا كَانَ لِيَأْخُذَ أَخَاهُ فِي دِينِ الْمَلِكِ إِلَّا أَن يَشَاءَ اللَّهُ نَرْفَعُ
دَرَجَاتٍ مَّن نَّشَاءُ وَفَوْقَ كُلِّ ذِي عِلْمٍ عَلِيمٌ﴾

صدق الله العظيم

سورة يوسف الآية ٧٦

Supervisor Certificate

I certify that thesis entitled "A Proposed Approach for ECG Signal Classification Using One Dimensional CNN" is prepared and written under my supervision at the department of Computer Science / College of Computer Science and Information Technology / University of Al-Qadisiyah as a partial fulfilment of the requirements of the degree of Master in Computer Science.

Signature: Lamia

Assist Prof. Dr. Lamia AbedNoor Mohammed

Date: / / 2021

Head of Department Certificate

In view of the available recommendations, I forward the thesis entitled "A Proposed Approach for ECG Signal Classification Using One Dimensional CNN" for debate by the examination committee.

Signature:

Dr. Qusay Omran Mosa

Head of the Department of Computer Science

Date: / / 2021

Certificate of the Examination Committee

*We, the undersigned, certify that (**Sarah Kamil Katfaan**) candidate for the degree of Master in Computer Science, has presented this thesis entitled (**A Proposed Approach for ECG Signal Classification Using One Dimensional CNN**) for debate examination. The examination committee confirms that this thesis is accepted in form and content and displays a satisfactory knowledge in the field of study based on the candidate demonstration during the debate examination held on: **30-November-2021**.*

Signature:
Name: Kadhim Brihi Swadi
Title: Prof. Dr.
Date: 8/12/2021
(Chairman)

Signature:
Name: Ali Mohsen Mohammed
Title: Assist Prof. Dr.
Date: / /2021
(Member)

Signature:
Name: Qusay Omran Mosa
Title: Dr.
Date: / /2021
(Member)

Signature: Lamia
Name: Lamia Abednoor Mohammed
Title: Assist Prof. Dr.
Date: / /2021
(Member)

Signature:
Name: Dhiah Edan Gabir
Title: Assist Prof. Dr.
Date: / /2021
(Dean of College of Computer Science and Information Technology)

Acknowledgments

All praise be to ALLAH Almighty who enabled me to complete this task successfully and my utmost respect to his last Prophet Mohammad PBUH and to the hero of Islam born in the heart of Kaaba Amir-Al-Mo‟mineen Al-Imam Ali Ibn Abi Talib.

First, all thanks, appreciation and respect to the wonderful woman who was with me step by step, my supervisor Dr. Lamia Abednoor Mohammed, for her invaluable guidance, supervision and untiring efforts during the course of this work.

I must express my very profound gratitude to my mother and my husband for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis.

I would like to declare my deep thanks and appreciation to my brothers, my sister and my friends for their patience, encouragement and help during the work.

Special thank goes to the staff of the Al- Qadisiyah university, especially the staff of the Presidency and staff of the college of Computer Science and Information Technology for the great help they have offered me...

Last but not least, but most importantly, I would like to thank all the kind, helpful and lovely people who helped me directly or indirectly to complete this work and apologize to them for not being able to mention them by name here, but they are in my heart.

Sarah Kamil Katfaan

Dedication

To the savior of mankind ...

Al-Imam Al-Mehdi

(may God hasten his honorable reappearance).

Who will fill the earth with justice and equity, after it has been filled with injustice and oppression. The survival of Allah in the earth.

To My Father ...

My soul, which left me alone long ago.

To my mother ...

You have always been a source of kindness and compassion; I grew up on the shoulders of your patience.

To my husband and children ...

Who are the secret of my happiness in life.

Abstract

Cardiovascular diseases (CVDs), including strokes and heart diseases, are a prominent type of life-threatening and chronic disease, and one of the leading causes of mortality worldwide. One of the most common CVDs is arrhythmia, which is a heart condition that occurs due to abnormalities in the heartbeat, which means that the heart's electrical signals do not work properly, resulting in an irregular heartbeat or rhythm and thus defeating the pumping of blood. In the clinical routine, cardiac arrhythmia detection is performed by electrocardiogram (ECG) signals, which is a non-invasive tool used for heart diseases diagnosis, particularly arrhythmia, by its signals which can reveal abnormal heart activity.

The diagnosis of arrhythmia in medical routine through ECG by checking beat-by-beat is considered a time-consuming and laborious action, in addition to the difficulty of visual interpretation of ECG signals due to their small amplitude and duration. Therefore, the automatic diagnosis of arrhythmia by the computerized algorithm can help the wearable devices be utilized as daily health monitoring systems, where reduced computing complexity and increased accuracy might result in correct diagnosis. More works have been suggested, which came *from machine learning and deep learning in order to achieve high performance. However, there is a challenge to reaching this goal.

As a result, we present a significant approach for identifying arrhythmias using ECG signals. In this study, a proposed approach based on Deep Learning (DL) technology that is a framework of nine-layer one-dimension Convolutional Neural Network (1-D CNN) for classifying automatically ECG signals into different cardiac conditions by using different classifications such as binary and multi-class classification.

In binary classification, the proposed CNN model detects only the normal from abnormal class while in the multi-class classification, seven different types of ECG signals named normal(N), left bundle branch block (LBBB), right bundle branch block (RBBB), atrial premature beat (APB), premature ventricular contraction (PVC), fusion beat (F), and Paced Beat (P) which are diagnosed using different classification schemes such as classification of three-, four-, six-, and seven-classes.

Practical testing of this work was carried out using the MIT-BIH Standard Database. The proposed model achieved the highest results in a three-class classification scheme that classify three different types of arrhythmia which are APB, LBBB, and RBBB with an average accuracy of 99.52%, a precision of 99.34%, a recall of 99.26%, a specificity of 99.76%, and a F1-score of 99.30% at training time of 361 s.

In the end, the proposed system was compared with some relevant modern models, and the achieved results showed that the proposed framework outperformed those models in most of the evaluation criteria, which indicates the success of the new approach's performance.

Keywords: *1-D CNN, Arrhythmia, Cardiovascular Disease, Deep learning, Detection and Classification, Electrocardiogram(ECG), Machine Learning, MIT-BIH arrhythmia database.*

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Meaning
AAMI	Association for Advancement of Medical Instrumentation
Acc	Accuracy
Adam	Adaptive Moments
AFDB	MIT-BIH Atrial Fibrillation Database
AFIB	Atrial Fibrillation
AFL	Atrial Flutter
ANN	Artificial Neural Network
APB	Atrial Premature Beat
APC	Atrial Premature Contraction
BGD	Batch Gradient Decent
BP	Backpropagation
CM	Confusion Matrix
CNN	Convolutional Neural Network
CUDB	Creighton University ventricular tachyarrhythmia Database
CVD	Cardiovascular Disease
CWT	Continuous Wavelet Transform
DL	Deep Learning
DNN	Deep Neural Network
ECG	Electrocardiogram
F	Fusion beat
FAV	First degree AV block
FFNN	Feed Forward Neural Network
FL	Focal Loss
FN	False Negative

FP	False Positive
GD	Gradient Decent
II-HB	Second-degree Heart Block
IV	Idioventricular rhythm
J	AV Junctional rhythm
LBBB	Left Bundle Branch Block
LSTM	Long-Short Term Memory
MITDB	MIT-BIH Database
ML	Machine Learning
MLII	Modified Limb Lead
MLP	Multi-Layer Perceptron
MV	Malignant Ventricular
N	Normal
NSRDB	Normal Sinus Rhythm Database
P	Paced Beat
Pre	Precision
PVC	Premature Ventricular Contraction
Q	Unknown beat
RBBB	Right Bundle Branch Block
ReLU	Rectified Linear Unit
RMS Prop	Root Mean Squared Propagation
ROC	Receiver Operating Characteristic
S	Supraventricular ectopic beat
SAV	Second degree AV block
Sen	Sensitivity
SGD	Stochastic Gradient Decent

Spec	Specificity
SVT	Supraventricular tachyarrhythmia
Tanh	Tangent hyperbolic function
TPR	True Positive Rate
TN	True Negative
V	Ventricular ectopic beat
VB	Ventricular Bigeminy
VEB	Ventricular Escape Beat
VF	Voltage Foot
VFib	Ventricular Fibrillation
VL	Voltage Left arm
VR	Voltage Right arm
VTR	Ventricular trigemini
VFW	Ventricular Flutter Wave
VT	Ventricular Tachycardia
WHO	World Health Organization

Chapter 1 - Introduction

1.1 Overview

Cardiovascular diseases (CVDs) are considered to be one of the leading causes related to human mortality. According to the World Health Organization (WHO) estimate calculation in 2017, CVDs are the major cause of death in the world. Over 17 million people die due to CVDs annually, which is about one-third of all deaths worldwide and with about 75 percent of the total CVD deaths occurring in countries with low and middle incomes. As a result, the detection and efficient therapy of CVDs are the most essential functions of all health care systems [1] [2] [3].

Cardiac arrhythmia is one of the most common types of CVDs. It is a range of disorders that affect the heart and circulatory system. In this condition, heartbeats vary from their normal pattern and the heart beats irregularly. A typical heartbeat fluctuates depending on age, activity level, body size, and emotions. Palpitations are a condition in which the heartbeat seems to be excessively rapid or slow [4].

The electrocardiogram(ECG) signal is a primary tool for predicting and diagnosing heart diseases. It is a vital modern medical instrument that records cardiac excitability, transmission, and recovery [5].

In clinical routine, a detailed examination of ECG signals gives functional information about the patient's heart, which has been widely employed for arrhythmia diagnosis. Because manually analyzing ECG features is time-consuming and laborious, it is absolutely essential to design automatic ECG analysis algorithms [6].

In recent years, several machine learning approaches in the domain of arrhythmia diagnosis automatically have been proposed in an attempt to

simplify the analyzing work of ECG features and address this problem. The typical procedure of these approaches is basically based on the three steps to evaluating ECG characteristics, which are

- preprocessing
- ECG features extraction
- classifying ECGs into different conditions using these features [7].

Deep neural networks (DNNs), which is a subset of machine learning, have been used to analyze ECG signals. It is one of the most effective technologies for diagnosing and predicting diverse types of serious diseases, particularly cardiovascular disorders. It can identify features in ECG data. Its layers can be learned to identify features in ECG signals without having to extract handcrafted features, with results that outperform clinicians[4][8] [9].

In this thesis, ECG data was obtained from the benchmark MIT-BIH arrhythmia database, it is open source database that hosted by physio net and included 48 recordings from 47 subjects [10]. A one-dimensional convolutional neural network (1-D CNN) is proposed for different classification schemes which are binary and multi-class classification. With binary classes, the proposed model has detected the arrhythmia class, while, with multi-class classification, the proposed model has used multiple for classify three, four, six, and seven classes of arrhythmia.

1.2 Thesis challenges

Some difficulties that make the arrhythmia detection and classification system unable to classify the arrhythmia classes with reasonable performance exist such as there are no optimum ECG signals classification criteria. Also, the imbalanced category is a major challenge to the arrhythmia

classification system, which results in a lower performance than the balanced category classification. ECG Arrhythmia classification may also face the challenge of developing the most suitable classifier that can classify arrhythmias in a reasonable time.

1.3 Thesis motivations

There are some motivations that encouraged the researchers to overcome the challenges of arrhythmia classification application to detect and classify multiple types of arrhythmia, some of these motivations are Summarized as the following:

1. classifying classes of arrhythmia that are threaten-life, causing cardiac arrest and sudden mortality. As a result, identifying and treating patients with life-threatening cardiac arrhythmias requires early automated detection and categorization of ECG abnormalities.
2. The classifying of multiple types of arrhythmias automatically in clinical routine by using computerized system can save time and effort, in addition to high accuracy.

1.4 Objective of thesis

The objective of this thesis is to design a new, efficient, and non-complex classifier model based on a deep learning approach for automatic classification that can classify various types of cardiac arrhythmias based on an ECG signal dataset with high performance.

1.5 Thesis contributions

The proposed system can assistance in the medical field by remotely monitoring the electrical activity of a patient's heart over time periods in

different lying environments such as hospitals, while traveling, at school or at home to detect and classify arrhythmias and thus save patients' lives.

1.6 Related works

Rapid advances in deep learning (DL) approaches to address diverse medical issues have provided medical sector experts with unparalleled support in many medical aspects, especially cardiology diagnosis. Recently, the diagnosis automatically of CVDs, especially arrhythmia, has been the most important aspect that has attracted the attention of researchers in the field of artificial intelligence.

In the past few decades, many research approaches have been proposed for exploiting ECG signals in arrhythmia detection and classification automatically using deep learning techniques. among these related works that have been recently published are:

1. Zubair et al., (2016) [11], the authors have presented deep learning approach based one dimensional convolutional neural network(1-D CNN) model with 4-layer for classifying ECG beats automatically into five classes: normal beat (N), supraventricular ectopic beat (S), ventricular ectopic beat (V), Fusion beat (F), and Unknown beat (Q). The classification performance of the proposed algorithm is evaluated using MIT-BIH database. the model has achieved accuracy of 92.7%. The model showed results that are not relatively high, in addition to it is not used other metrics to illustrate the actual performance of it.
2. Acharya, Fujita, Lih, et al., (2017) [12]**,** the authors have employed 11-layer 1-D CNN model for classification of normal (N) and three classes of tachycardia arrhythmias that are, atrial fibrillation (AFIB), atrial flutter (AFL), and ventricular fibrillation (VFib) with ECG segments of two-second (21,709 segments) and five-second (8,683 segments). The

network with two-second ECG segment has achieved results as follows: an accuracy of 92.50%, specificity of 93.13%, and a sensitivity of 98.09%. While, the results that it has achieved with five-second ECG segment are: accuracy of 94.90%, specificity of 81.44%, and sensitivity of 99.13%. The databases that were used in this study are MITDB, MIT-BIH Atrial Fibrillation database(AFDB), and Creighton University Ventricular Tachyarrhythmia database(CUDB).

3. Acharya, Oh, et al., (2017) [13], the authors have developed a 9-layer CNN model for the classification of ECG signals from MITDB automatically by identifying five main categories of arrhythmias of non-life-threatening that are: Ventricular ectopic (V), Supraventricular ectopic (S), Fusion ectopic (F), Unknown ectopic (Q), and Non-ectopic (N) . Two set of experiments are applied that are:
   - The first set (set A) contains ECG signals with noise removal and the achieved performance is 93.47% accuracy, 96.01 sensitivity, and 91.64% specificity.
   - The second set (set B) contains ECG signals that have not been filtered for noise (original data). The achieved performance of the model with this experiment are accuracy of 94.03%, sensitivity of 96.71%, and specificity 91.54%.

   This approach has certain disadvantages, such as the need for extended training hours, specialized hardware (Graphic Processing Unit), computationally expensive training, and a large number of data for training.

4. Isin & Ozdalili, (2017) [14] in this work, an 8-layer Alex Net has used as a transferred deep learning framework which was previously trained

on a generic imaging dataset. This model is applied as a features extractor; the extracted features are input into a basic back propagation neural network to complete the final classification that classifies the patient's ECGs into relevant cardiac diseases that are normal (N), right bundle branch block (RBBB) or paced beats (P). To assess the proposed framework, the relevant conditions of ECG waveform are chosen from MITDB. This system has obtained a sensitivity of around 92% and a recognition rate of 98.51%. This model has the benefit of being simple to use since it eliminates the need to train a deep convolutional neural network. It also has some limitations, such as there are only a few types of ECG arrhythmias to classify and the model uses many filters, which leads to ECG beat losses.

5. Oh et al., (2018) [15], have proposed an automated system that combined Long Short Term Memory (LSTM) and CNN algorithms for classifying five classes of ECG beats with variable length named as : normal (N), premature ventricular contraction (PVC), right bundle branch block (RBBB), left bundle branch block(LBBB), and atrial premature beat(APB). For this model, ECG dataset has downloaded from MITDB and the results that gained from this classifier are accuracy of 98.10%, sensitivity of 97.50%, and specificity of 98.70%**.**

6. Savalia & Emamian, (2018)[16], have suggested a comparison between MLP (multi-layer perceptron) and CNN as a classification system for heartbeats. By MLP, two classes (arrhythmia and normal) are detected, while by using 4-layer CNN, nine classes are classified which are: arrhythmia, first degree AV block (FAV), second degree AV block (SAV), atrial fibrillation (AFIB), atrial flutter (AFL), malignant ventricular (MV), normal sinus (N), ventricular tachycardia (VT), and ventricular bigeminy with evaluation performances of 88.7% and 83.5%

for MLP and CNN, respectively. In this method, despite having the ability to classify many types of arrhythmias using a simple architecture, it has the drawback of having low classification performance that makes it difficult to use in practice.

7. W. Zhang et al., (2018)[7], have presented a classifier based on 12-layer one dimensional CNN for classifying one lead individual heartbeat signals into five different classes of arrhythmia which are: Normal beat (N), Left Bundle Branch Block (LBBB), Right Bundle Branch Block (RBBB), Premature Ventricular Contraction (PVC) beat, and Atrial Premature Beat (APB). This model has evaluated on the MITDB and the performance obtained as precision of 0.977, F1-score of 0.976, and the sensitivity of 0.976.
8. Yildirim et al., (2018) [17], the authors have presented 16-layer 1-D CNN model for detection 17 classes of cardiac arrhythmia based on the analysis of long-duration(10-s) ECG signals fragments from MITDB. In addition to the original dataset that comprised 17 classes, two additional sub-datasets with 13 and 15 classes were produced. The achieved accuracy in the classification of 13-class, 15-class, and 17-class was 95.20%, 95.20%, and 91.33%, respectively. This model has a limitation that is includes many layers which increases its computational complexity.
9. Hua et al., (2019) [18], the authors presented 7-layer 1-D CNN to classify 5 classes and 6 classes from 48 recordings from MIT-BIH and 5 classes from 48 recordings by Association for Advancement of Medical Instrumentation (AAMI) with accuracy of 99.24% and 97.02%, sensitivity of 99% and 97% , and F1-score of 99% and 97% for 5 classes and 6 classes from MIT-BIH, respectively. The classifier

also achieved a performance of 97.45% accuracy, sensitivity and F1-score of 97%.

10. Avanzato & Beritelli, (2020) [19], the authors have applied 5-layer 1-D CNN to classify three types of ECG signals, named normal, premature ventricular contraction, and atrial premature beat, several metrics are used for measuring the achieved performance which are average accuracy, sensitivity, and F1-score of 98.33%, specificity of 98.35%.
11. Wan et al., (2020) [20], have used 4-layer 1-D CNN to classify five types of kinds of ECG data which are normal, ventricular premature beat, left bundle branch block, right bundle branch block, and paced beat. The experimental verification is carried out on the data from MITDB with accuracy of 99.10%.
12. Petmezas et al., (2021) [21], have proposed a hybrid system of one dimensional CNN and LSTM with focal loss (FL) to deal with ECG datasets imbalance. The model has been trained on the MIT-BIH atrial fibrillation database where four classes of ECG rhythm has classified which are normal, atrial flutter, AV junctional rhythm, and atrial fibrillation, with an achieved performance of 99.29% specificity and 97.87% sensitivity.
13. Ullah et al., (2021) [22], have used five layers 1-D CNN and seven layers 2-D CNN for classifying five classes and eight classes of ECG signals, respectively. The achieved performances with this study is a classification accuracy of 97.38% and 99.02% for 1-D and 2-D CNN, respectively. In this study 1-D dataset are downloaded from MITDB for 1-D CNN and the same data are transformed in to 2-D images for the 2-D CNN by continuous wavelet transform (CWT).

Related works are summarized in table (1-1)as follows:

Table (1-1): Related works summary

N0.	Model	Preprocessing	Classes	Database	Performance
1	4-layer 1-D CNN	• R-peak detection	N, F, S, V, Q	MITDB	Acc = 92.70%
2	11-layer 1-D CNN	• Segmentation • Z-score normalization	N, AF, AFL, VFib	MITDB AFDB CUDB	Net A (with two ECG segment): Acc = 92.50% Sen=98.09% Spec =93.13% Net B (with five ECG segment): Acc = 94.90% Sen = 99.13% Spec = 81.44%
3	9-layer CNN	• R-peak detection • Noise removal • Segmentation • Z-score normalization	N, S, V, F, Q	MITDB	with noise removal: Acc = 93.47% Sen = 96.01% Spec = 91.64% without noise removal: Acc = 94.03% Sen = 96.71% Spec=96.71%
4	8-layer Alex Net	• Noise removal • QRS detect	N, P, RBBB,	MITDB	Sen = 92% Acc = 98.51%
5	CNN+LSTM		N, PVC, LBBB, RBBB, APB	MITDB	Acc=98.10% Sen = 97.50% Spec = 98.70%
6	5-layer 1D CNN/MLP		Arrhythmia, N, MV,SAV, FAV,VT, AFL, AFIB BIGEMINY	MITDB NSRDB	Acc of 88.7% MLP 83.5% CNN
7	12-layer 1D-CNN	• WT • Z-score normalization	LBBB, PVC RBBB, APB	MITDB	Pre = 97.7% Sen = 97.6% F1-score=97.6%
8	16-layer 1D-CNN	Rescaling raw data	N, APB, VB AFL, AFIB, SVT, WPW	MITDB	Results on 13 classes: Acc=95.20%

			Pre-excitation, PVC, VTR, IV, VT, VFib, F, LBBB, P RBBB,II-HB		Results on 15 classes: Acc = 92.51% Results on 17 classes: Acc = 91.33%
9	7-layer 1-D CNN	R-R-R segmentation	N, LBBB, RBBB, V, P, extra class, S, F, Q	MITDB	Results on 5 classes: Acc= 99.24% Sen= 99% F1-score= 99% Results on 6 classes: Acc= 97.02% Sen= 97% F1-score= 97% Results on 5 classes from AAMI: Acc= 97.45% Sen=97% F1-score= 97%
10	5-layer 1D CNN	Without	N, PVC, APB	MITDB	Acc = 98.33% Sen=98.33% F1-score=98.33% Spec = 98.35%
11	4-layer 1D CNN	• Linear filtering • R-wave detection	N, PVC, LBBB, RBBB, P	MITDB	Acc= 99.10%
12	8-layer 1D CNN–LSTM	• Focal loss • DE noising • R-peak detection	N, AF, AFL, J	AFDB	Sen = 97.87% Spec = 99.29%
13	5-layer 1D-CNN 7-layer 2D-CNN	Noise removal	Classes with 1D-CNN: N, PVC, APC, LBBB, LBBB Classes with 2D-CNN: N, VFW, P, PAC, PVC, VEB,RBBB, LBBB	MITDB	For 1D –CNN: Acc = 97.38% For 2D-CNN: Acc = 99.02%

1.7 Thesis Organization

This dissertation comprises five chapters. The present chapter (Introduction) introduces a comprehensive introduction to provide the reader an insight into the research problem, motivations to overcome it, research's objectives and contributions, as well as some relevant studies that attempted to solve the problem using other methods on the same datasets. The rest of the thesis is structured as the following:

- Chapter two (Theoretical Concepts) reviews an extensive description of the main concepts of arrhythmia, ECG signals, deep learning, and evaluation techniques.
- Chapter three (Proposed Work) presents the general system architecture as well as the technical materials used, data preparation (preprocessing methods), and training setting.
- Chapter four (Experiments and Results) presents a description of the data used and explains the achieved results obtained from applying the proposed model to the employing dataset as well as their evaluation.
- Chapter five (Conclusions and Future Works) presents the main Thesis' conclusions as long with recommendations for future works.

Chapter 2 – Theoretical Concepts

2.1 Introduction

In this chapter, some theoretical concepts that based in the hypothesis of this thesis, are described. First, the definition of the human heart is given as well as its main functions and diseases. Furthermore, emphasis has been given to cardiovascular diseases, arrhythmias and their common types. Moreover, focusing on the electrocardiogram (ECG) signal, its main structure, and the most important features of this type of signal. Then, the concepts of artificial neural networks (ANNs) and deep learning, such as their structures and parameters, are introduced. Also, several machine learning performance evaluating techniques, such as the basic evaluation measures, Receiver Operating Characteristic (ROC) curve, as well as confusion matrix are presented. Finally, normalization preprocessing step has been illustrated.

2.2 Human heart and arrhythmia

The human heart and the diseases that are associated with it, especially arrhythmia, is a vital and important issue due to its great impact on human life, as it will be explained in the following sub sections.

2.2.1 Heart of human

The heart is a muscular organ that maintains two principal functions:

I. Collected the blood from the lungs and pumped it to all of the body's tissues.
II. Pumped back the blood that is collected from the body’s tissues to the lungs [23].

The heart's internal anatomy reveals that the heart muscle or myocardium is composed of four main chambers, which are the two upper and lower chambers. The main function of the two upper chambers, or atria, is to collect

blood. While the lower chambers, or ventricles, are more powerful and their main function is to pump blood [23].

2.2.2 Cardiovascular Diseases

Cardiovascular diseases (CVDs) have considered as a major issue that threatens the safety of human life. Because of the high risk of sudden mortality, early detection is important and more urgent than therapy[24]. CVDs are the primary cause of death worldwide, according to statistics from the World Health Organization (WHO) [25] [26]. Fig (2.1) illustrates the ratios of death causes, which compared the heart diseases with other death causes ratios for low- and middle-income regions.

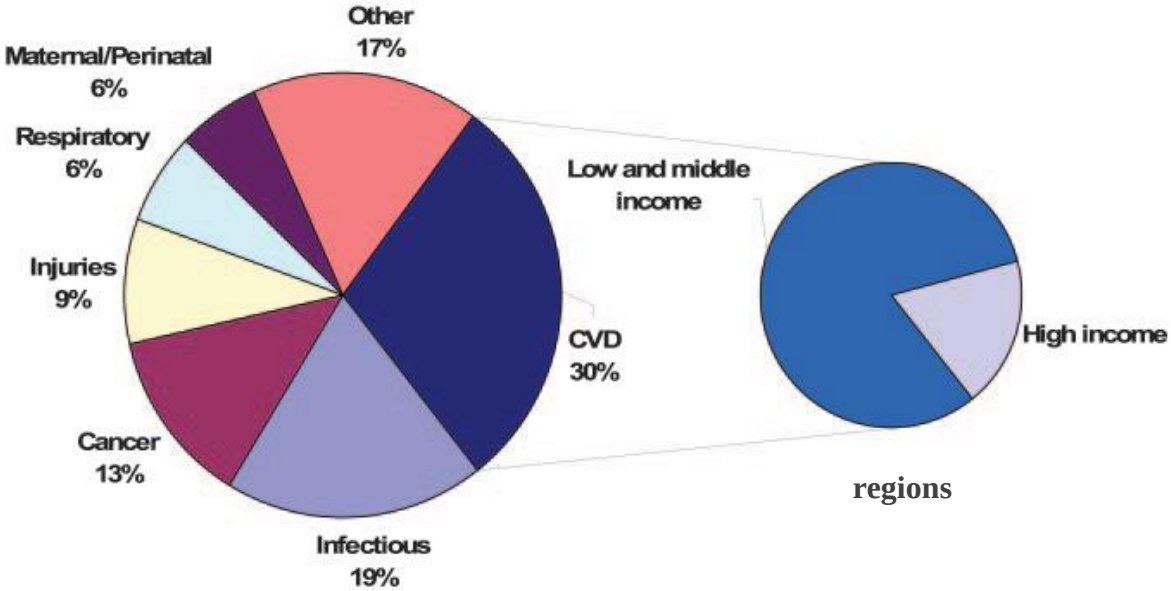


Fig. (2.1): Cardiac diseases compared with other factors of mortality [26].

2.2.3 Arrhythmia

Arrhythmia is a cardiac condition that is associated with abnormalities in the heartbeat or rhythm. As a result, the human heart beats slowly, rapidly, or irregularly. The heart may be unable to pump a sufficient amount of blood into the body during an arrhythmia. The heart, brain, and many other organs

can be damaged by decreased blood flow and every year, approximately half a thousand people in the United States die due to cardiac arrhythmias. [27][28].

Cardiac arrhythmias are divided into two categories based on their severity: life-threatening and non-life-threatening arrhythmias. The first kind has the potential to cause cardiac arrest and death. While the other may not always lead to heart failure, it does need prompt treatment to prevent further decline in heart function [28]. On the other hand, arrhythmias are divided into two main groups. The first group includes morphological arrhythmias, which are arrhythmias caused by a singular irregular heartbeat, while the other type is rhythmic arrhythmia, which is created by a series of irregular heartbeats [29].

A physician can use the fiducial points along the ECG to measure the time intervals and this, in turn, determines the irregular activities of the heart. The type of arrhythmia and which part of the heart is hypertrophic can be determined by a physician by measuring the amount of waves that move across the heart [30].

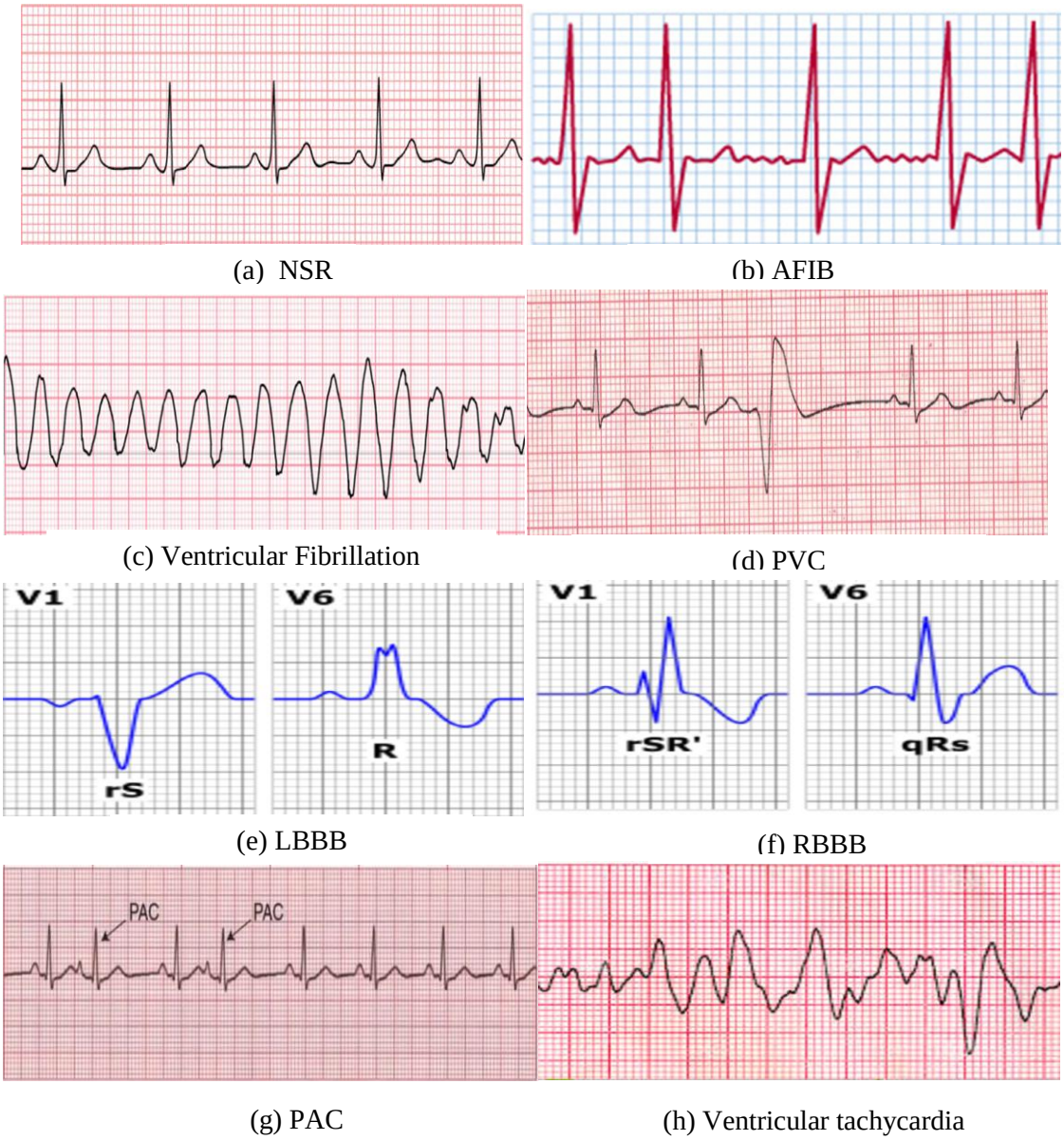
2.2.4 Types of arrhythmia

There are several types of arrhythmia such as:

1. Normal sinus rhythm(NSR) as shown in fig. (2.2) (a) it is any cardiac rhythm in which the heart rate ranges from 60 to 100 beats per minute[31].
2. Atrial fibrillation (AFIB) as shown in fig. (2.2) (b), it is one of the most popular cardiac arrhythmias that occurs due to a rapid heart rate (more than 100 beats/min). It is more prevalent among the elderly than among the younger generation. AF has very dangerous health implications that

are associated with an increased risk of mortality, heart failure and stroke [3] [32].

3. Ventricular Fibrillation (VFib) as shown in fig. (2.2) (c), it is the main cause of sudden cardiac mortality, which is diagnosed on the ECG by the irregularity and aperiodic changes in the ventricular. The heart rate is too high (more than 550 excitations/minute) during VFib to enable sufficient blood pumping [33] [34].
4. Premature Ventricular Contraction (PVC) and premature atrial contraction (PAC) as shown in fig. (2.2) (d, g), respectively. PVC and PAC occur when the normal rhythm of the heart is disrupted by an early or premature beat. A PVC originates in the ventricles, while a PAC comes from the atria [35].
5. Left and Right Bundle Branch Block (LBBB, RBBB) as shown in fig. (2.2) (e,f), respectively, are defined as an aberrant QRS morphology caused by a disruption in the typical conduction system[35][36].
6. Ventricular arrhythmias as shown in fig. (2.2) (h), such as ventricular fibrillation and ventricular tachycardia, are a leading cause of morbidity and sudden cardiac death, which are responsible for more than four million deaths annually. Generally, ventricular arrhythmias can be caused by different problems, such as weakened heart muscle, heart attack, and coronary heart diseases [27] [37] [38].
7. Fusion and paced beats: fusion beats are a parasystolic state in which two different resources' electrical pulses act on the same area of the heart at the same time[35][39]. While, paced beat is a heartbeat which is generated by a pacemaker [40].



(a) NSR (b) AFIB

(c) Ventricular Fibrillation (d) PVC

(e) LBBB (f) RBBB

(g) PAC (h) Ventricular tachycardia

Fig. (2.2): A sample of arrhythmias types [41].

2.3 Electrocardiogram (ECG)

The Electrocardiogram (ECG) is a signal that is rich in a wealth of information that may be utilized for the detection and diagnosis a variety of heart disorders [42]. It is a non-invasive medical tool that is widely used due to its simple nature [29].

In the standard procedure, ECG is implemented by using ten electrodes that are attached to the skin surface in specific regions of the human body, as illustrated in the fig. (2.3). These electrodes produce twelve different signals known as LEADS, which are six chest leads (V1 to V6) and six limb leads (I, II, III, a VR, a VL, and a VF). The information from the twelve leads is grouped to perform an electrocardiogram [29][43].

ECG signal is made up of multiple ECG beats and each ECG beat has a P wave, QRS complex, and T wave. A typical ECG waveform has some ECG features, as shown in the Fig. (2.4), which include:

I. Six fiducial points or peaks (P, Q, R, S, and T)

II. Segments (ST and PR)

III. Intervals (PR, ST, and QT)

These features have typical duration and amplitude values, and any variance from these values can aid in the detection of a variety of abnormal cardiac disorders[24] [31][44].

Traditionally, on ECG, the heart rate has been predicted as the number of heartbeats per unit time and A typical heartbeat is between 60 and 100 beats per minute. Based on the average irregular heartbeat, there are two forms of arrhythmia, which are tachycardia and bradycardia, In tachycardia, the heart rate is too fast (more than 100 beats/min), while in bradycardia, the heart rate is too slow (below 60 beats/min) [45] [46][47].

There are many difficulties that can be seen as challenges in classifying the ECG, including: ECG waveforms in different patients that are different in amplitude, timing, and signal slopes as well as the lack of standard procedures for classifying ECG waveforms. As a result, ECG signals need a rigorous classification procedure as well as appropriate model selection [44].

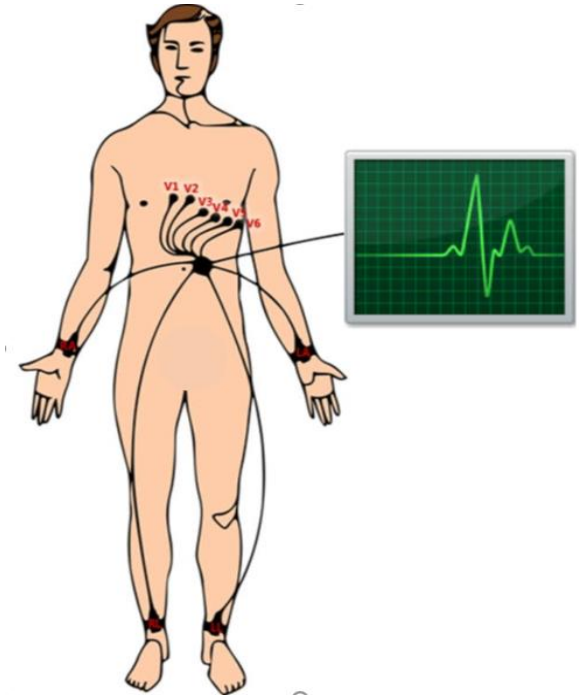


Fig. (2.3): The typical ten electrodes configuration [29]

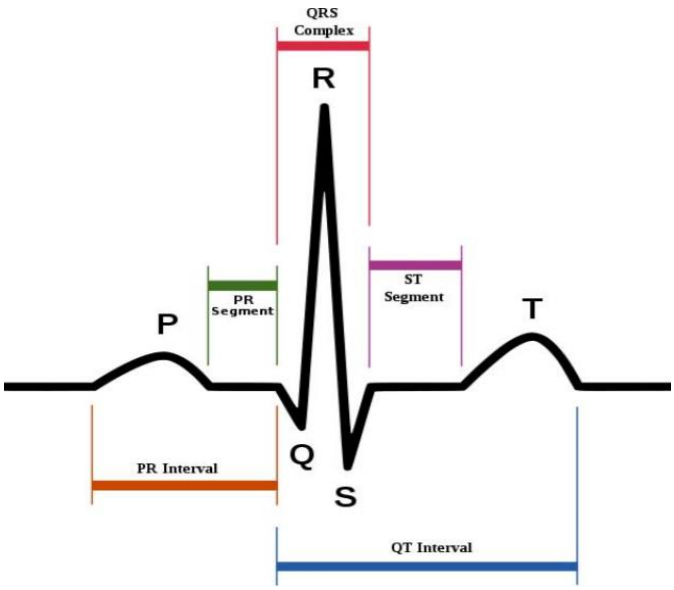


Fig. (2.4): A typical ECG waveform [24].

2.4 Artificial neural network (ANN).

An artificial neural networks (ANNs), also known as Neural Networks (NNs), are a machine learning approach inspired by the human brain's ability to complete complicated tasks using interconnected neurons that conduct

relatively simple operations on each other. Similarly, a neural network (NN) is a trainable structure which has an input layer followed by one or multiple hidden layers for mapping nonlinear relationships, and then an output layer of neurons. It also involves groups of connections with numeric values for linking the neurons called weights[8][48][49]. In the basic construction of an ANN, a neuron is the most fundamental element, a single artificial neuron is shown in fig. (2.5) [50].

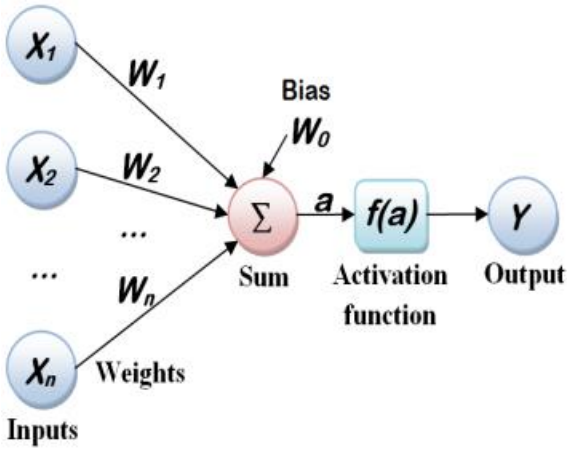


Fig. (2.5): Single artificial neuron [50]

ANNs have several common characteristics as follows:

1. They consist of large number of neurons which are linked units, each neuron takes inputs, analysis them, and then produces an output [51].
2. The units are connected in certain patterns and organized in specific designs.
3. The network is respond to data presented or perform a task by training rather than by programming [52].

Feed forward neural networks(FNNs) are a perfect example and one of the most widely used architectures of ANNs with layers of neurons and only forward connections. This results in a strong connectionist model that can learn any type of continuous nonlinear mapping, with applications as diverse as Time Series Forecasting, Handwritten Recognition, and Medical

Diagnostics [53]. A term ‘deep’ is called on the Feed forward neural network with multiple hidden layers [54].

2.5 Neural network’s parameters

There are several parameters that are implemented in NN as illustrated in the following subsections:

2.5.1 Cost function and gradient – descent

The cost function calculates the difference between the actual output and the target output of the neural network. Its value is minimized by adjusting the weights in the optimization process [8]. To assess the error in output data during back propagation, there are basically three types of cost functions which are:

I. Cross entropy cost function
II. Soft max loss function
III. Quadratic cost function

The Cross entropy cost function and Soft max loss function are commonly used in image classification and recognition, while the Quadratic cost function is well known in function regression [8].

The gradient descent (GD) is one of the most popular optimization algorithms in which play an extremely important role in the training the models of machine learning (ML) and deep learning (DL). The GD optimization algorithm is used to find the parameter values of a function which contribute to decreasing the loss function to the minimum possible level. despite their growing popularity, are frequently employed as black box optimizers due to the lack of practical explanations of their weaknesses and strengths [55][56]. Basically, there are three variants of the gradient descent

algorithm for finding the minimal value of the cost function [57], which are batch gradient descent (BGD) or also known as vanilla GD, stochastic gradient descent (SGD), and Mini-Batch gradient decent [58].

Root Mean Squared Propagation (RMS Prop) and Adaptive Moments (Adam) are the most often used adaptive stochastic algorithms. Adam is preferable due to the highest accuracy of the test results that it can be performed, it is a stochastic optimization approach that employs the SGD principle. On the other hand, the RMS prop optimizer that works with the root mean square value of the gradient change [59][60][61].

2.5.2 Backpropagation

The Backpropagation Network (BPN) is the most commonly used practical model of neural networks for monitoring learning. The BPN was first presented in 1974 by Verbose, and it owes much of its development to Rumelhart. BPN has dominated in the pattern recognition or classification area [62].

BPN is considered as a significant part that reduces the error at the time of the training phase. The method of backpropagation training involves some steps as follows:

1. Feedforward the input training pattern
2. Error calculation and backpropagation
3. Adjusting the weight values [63].

The general back propagation neural network is illustrated in the fig. (2.6) [64].

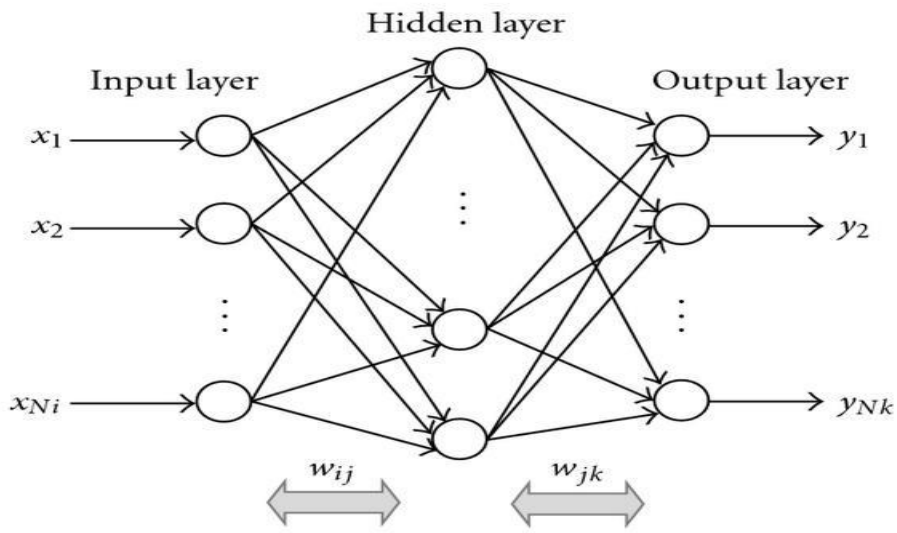


Fig. (2.6): Backpropagation neural network [64].

2.5.3 Overfitting and Under fitting

Overfitting is a fundamental issue in the applications of supervised machine learning. This phenomenon is determined when the training dataset is fitted by a learning algorithm so well noise and training data's peculiarities are retained. Overfitting causes a problem of poor generalization for the model because the model does not well generalize from observable to unknown data on the training and testing sets [65][66].

Besides the presence of noise, the overfitting may be happening due to the complexity of the classifiers and the limited size of the training set. In the overfitting model, all the features are taken into consideration, even if some of them have a minor impact on the final outcome. Worse, some of them are noises that have no impact on the output, and, as a result, the model becomes complicated[66].

The opposite of overfitting is under fitting. This happens when the model is unable to capture the data's variability [65]. Preventing overfitting is one of the basic challenges in the training of Deep Neural Networks

(DNNs), Many techniques for minimizing or preventing overfitting without a significant quantity of training data have been presented. The strategies are [66] [67].

1. Regularization
2. Early stopping
3. Network reduction
4. Expansion of training data (data-augmentation)

Regularization is the most popular strategy, which is presented to ensure the performance of the models to a significant degree when dealing with real-world situations by depending on the feature selection principle and distinguishing the more useful features from the less useful or useless features. The general regularization methods are L1 regularization, L2 regularization, and Dropout [66]. Overfitting may also be reduced by training the data on big datasets. If a dataset is limited, data augmentation techniques, which is efficient strategy for generalizing the model and reducing overfitting, can be used to artificially expand it. It enhances the training data diversity of the model and it becomes less dependent on the given data [61] [68].

2.5.4 Learning phase

The learning process of ANNs has a significant impact on their efficiency. The training of the CNN architecture uses the backpropagation algorithm, which is a common example of a gradient-based method, as shown in the fig (2.7). The training process involves two main steps, which are propagation and weight updating. The propagation step has involved feed-forward and back propagation passes [69].

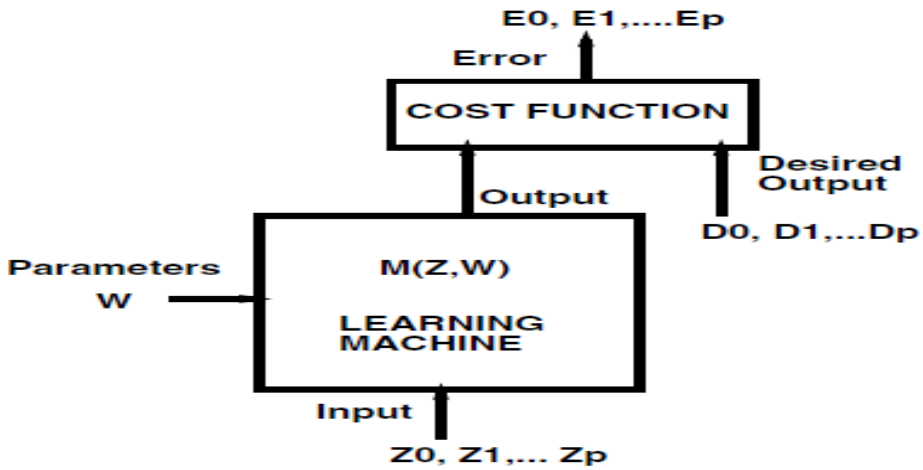


Fig. (2.7): The gradient-based learning machine [70].

In the feedforward pass, the feature maps are determined by the input vector by passing from left to right across the layers until reaching the output layer. In the backpropagation pass, the errors are propagated from right to left and the chain-rule is used for calculating the derivatives of the predicted errors on each layer. The backpropagation uses the cost function to calculate the propagation errors for the predicted output. The cost function is defined in equation (2.1) as:

$$E^P = \frac{1}{2}(D^P - M(Z^P, W))^2 \qquad (2.1)$$

Where $M(Z^p, W)$ is a function where Z^P is the pth input pattern, W is the adjusted parameters, and D^p is the desired or target output for the pattern Z^p. The average cost function is the average of errors over the training set as defined by the equation (2.2) as:

$$E_{train=} \frac{1}{p}\sum_{n=1}^{s} E^P \qquad (2.2)$$

Where n is a number of examples. In the second step, the weights are updated as follows: the input activation and the weight's output delta are multiplied to find the weight's gradient and its learning rate is subtracted

from the weight. This cycle of learning is repeated until the errors within the network are reached at a satisfactory value [71].

2.5.5 Hyper parameters and optimization

The meta-parameters (also known as hyper parameters) and the training parameters are two sets of parameters utilized by all machine learning algorithms. The parameters' values for each set were specified at the corresponding phase [72]. The algorithm that may be utilized in the learning process will be determined by hyper parameters. Various model training algorithms require distinct hyper parameters, which influence the outcome of the learning process. A number of hyper parameters influence the performance of neural network classifiers such as learning rate, depth, and batch size [60][73].

The learning rate is one of the most important hyper parameters which defines how many steps it will take to reach the global minimum. Generally, a training algorithm with a low learning rate would converge slowly, whereas a training method with a high learning rate will diverge [61][74].

The number of epochs and the batch size are other examples of hyper parameters, which are both integer numbers that appear to perform the same function. The batch size determines how many training samples or instances the model must go through before its internal parameters are changed. While the number of epochs determines how many full passes over the training dataset are made [75].

The optimization of the hyper parameters is a method for finding the perfect hyper parameters group which makes the generalization error of the given learning algorithm is shallow [76]. In deep neural networks (DNNs),

the optimization process of the hyper parameters is often done by Bayesian optimization, grid search, or random search, and the two latest methods are the most popular. Grid search takes a long time because it searches for the perfect model from a large number of models based on preset hyper parameter values, while random search takes less time than grid search by selecting models randomly with various combinations of hyper parameters [77] [78].

2.5.6 Activation function

The activation function, also known as the transfer function, is an essential part of the artificial neural network. It is used in a particular layer to convert input data into output by applying it to the sum of the product of the inputs and their weights to produce output which is then fed to the next layer in the hierarchy as input. In neural networks, the absence of an activation function simply makes the output signal a linear function that is polynomial of first degree [79][80].Depending on the function they represent, the activation functions are divided into two basic types [81], these types are:

1. Linear function whose output value is the same as its input [64]. The general equation for the linear function has shown in equation (2.3):

$$y = f(x) = x \quad with: f'(x) = 1 \qquad (2.3)$$

2. Non-linear activation functions such as sigmoid, Tanh, and RelU functions [82] are illustrated in the next subsections as follows:
   - The sigmoid function is one of the most common non-linear activation functions that transforms the output of neurons into the range [0:1][80]. The sigmoid function has illustrated in equation (2.4):

$$f(x) = \frac{1}{1 + e^{-x}} \qquad (2.4)$$

- The Tanh function, also known as the Tangent Hyperbolic function, is similar to or can be derived from the sigmoid function. The values in the Tanh function lie in the range of [-1:1] [80]. The Tanh function can be illustrated in the following equation (2.5):

$$f(x) = 2sigmoid(2x) - 1 \quad (2.5)$$

- The Softmax function is used in neural networks as output activation function for predicting categorical probability distribution [83]. Softmax function can be illustrated in the equation (2.6) as:

$$f_j(z) = \frac{e^{z_j}}{\sum_{k=1}^{n} e^{z_k}} \quad (2.6)$$

where z is an arbitrary vector formed at the ith layer of the CNN with real values z_j, j = 1,…., n, and n is the vector's size [84].

- The Rectified Linear Unit (ReLU) is the most commonly used among the nonlinear activation functions, which has a positive impact on the various tasks related to machine learning. [85] [86]. It sets the positive values unchanged and discards the negative values or sets them as zero. [86], RelU is presented in equation (2.7)(2.8) as:

$$f(x) = \max(0, x) \qquad (2.7)$$

$$Relu\ (x) = \begin{cases} x, & if\ x \geq 0 \\ 0, & if\ x < 0 \end{cases} \quad (2.8)$$

The ReLU is the default option activation function in the deep learning area due to its effectiveness and simplicity [87]. One of the shortcomings of ReLU is that it causes degradation of the network performance by treating the negative values as not found or unimportant [86].

2.6 Deep learning

Deep learning (DL) is a subfield of machine learning (ML) under artificial intelligence (A.I) [88]. DL has recently acquired prominence, owing to notable accomplishments in natural language processing, reinforcement learning, and image classification [89]. The structure of DL is extending the traditional neural networks (NNs) by putting additional hidden layers to the network design between the input and output layers to model nonlinear (as shown in the fig. (2.8)) and more complex relationships [90]. DL has many great advantages, such as extracting features automatically from the raw data which eliminating the necessary for the manual manner and speeding up the solving of complex problems, in particular well [91] [92] [93].

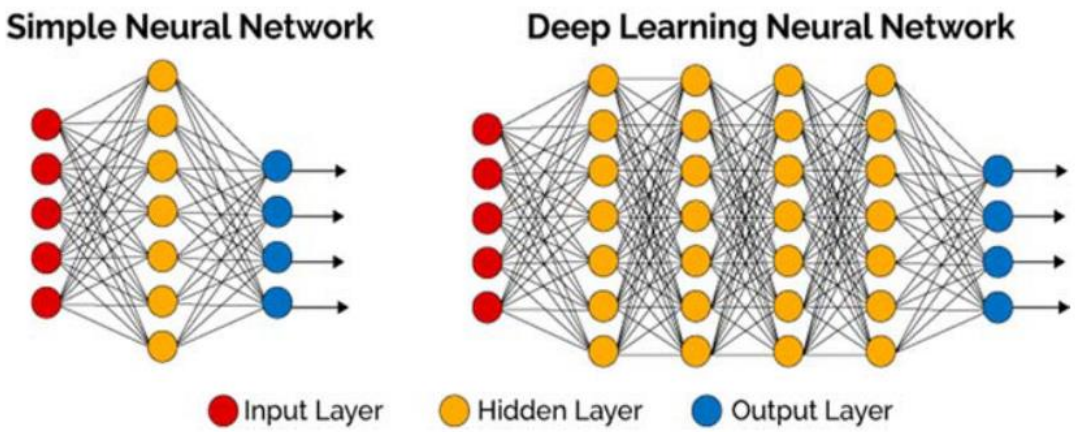


Fig. (2.8): comparison between a simple and deep learning neural network [16].

As a comparison between DL and machine learning models, DL models provide a number of clear advantages over conventional machine learning models, including the use of comprehensive feature engineering. Classical ML algorithms rely on manually extracted features, which takes time and results in over-specified and incomplete features. DL, on the other hand, needs a large amount of processing power and storage owing to the increased

number of hidden layers compared to traditional ML algorithms, but the accuracy gains are larger [94].

2.7 Convolutional neural network (CNN)

The convolutional neural network (CNN) is one of the most popular types of neural network that was introduced by LeCun in the 1990s to recognize handwritten digits. CNNs are models of machine learning that are based on supervised learning [93][95][96][97]. It is a deep learning algorithm that extracts the features by convolving the kernel or filter with input images [61].

Generally, the typical structure of CNN as shown in fig. (2.9) involves a set of convolutional and pooling layers that assist in extracting meaningful features from the image maps, in addition to flatten, dropout, fully-connected, and output layers [98].

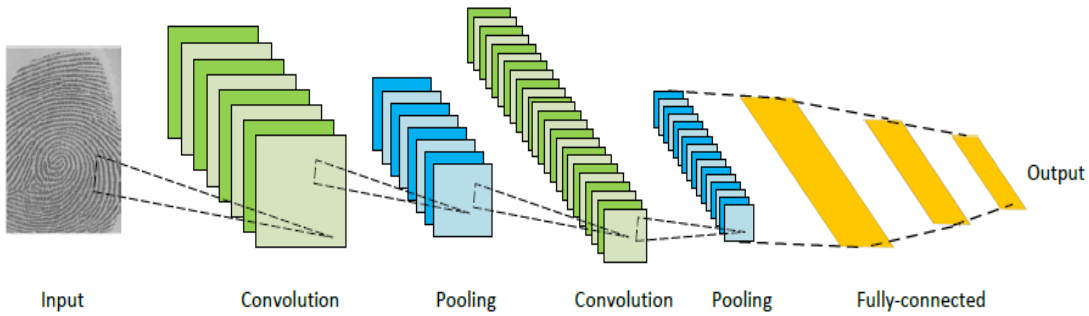


Fig. (2.9): Convolutional neural networks' architecture [97].

2.7.1 Convolutional layer

Convolution layer is a primary construction block in the CNN that responsible the majority of the computationally difficult lifting. The main goal of this layer is to extract the features from the inputs [99]. It involves a set of feature maps. Neurons in the same feature map have the same weights

(also known as kernels or filters), but they receive distinct frequency-shifted inputs [100].

The essential idea of this layer revolves around a linear operation known as convolution, which performs a filtering operation process by applying a filter (or kernel) to the input to produce the feature maps. The spatial dimensionality of the kernels is generally low, yet they cover the whole input's depth [101] [102]. The kernel size is also known as the filter size that convolves all over the feature map. Whereas the amount of the kernel that has been slide around the feature map is called the stride, it refers to how much the filter moves [99].

The kernel is represented by square matrices such as 2x2 or 3x3 dimensions. The kernel is applied over a region of the input map that is equal to the size of the kernel. The kernel slides from left to right and then from top to bottom, from the first pixel in the top left portion to the last pixel in the right bottom portion of the input map. This process is called the convolution operation, as illustrated in the fig. (2.10).

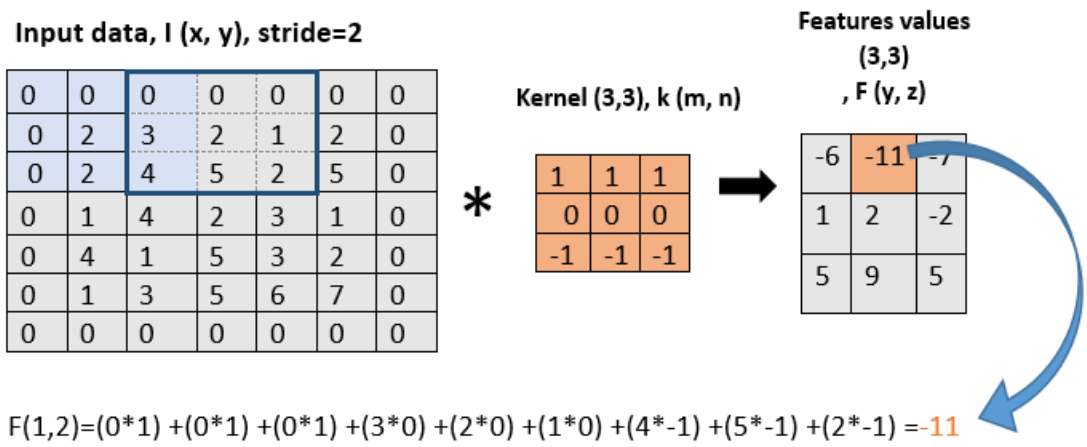


Fig. (2.10): The Convolution operation for feature value extraction

For the convolutional layer, if x_i^0=[x_1,x_2,…,x_n], is the data input vector for beat samples; where n is the samples' number per beat, then the general equation for calculation the output can be represented in equation (2.9) as shown below:

$$C_i^{l,j} = f(\sum_{m=1}^{M} w_m^j \; x_{i+m-1}^{0j} + b_j) \quad (2.9)$$

Where $C_i^{l,j}$ is the output of the convolutional layer; l is the layer index; f *is* the activation function that is typically a ReLU, sigmoid, or hyperbolic tangent (TanH) function; M is the size of the kernel/ filter; m is filter index and $w_m^j$is the weight for m^{th} filter index and j^{th} feature map; input data ; b_j is the bias for the jth feature map[11].

2.7.2 Pooling layer

The pooling or subsampling layer makes the features extracted more robust against distortion and noise by reducing the feature resolution [103]. It is used to modify the output of feature maps by reducing their spatial dimensionality, and thus leads to a variety of benefits, including regularizing overfitting, accelerating computation, and minimizing the parameters number [104].

There are several pooling methods and the most popular are max pooling and average pooling. In the max pooling method, the image is partitioned into sub-regions in the shape of rectangles and then returned only the maximum value within these sub-regions, while the average pooling method calculates and returns the average value within these sub-regions [105].

For example, the output layer in the case of a pooling size of 2*2 and an input feature of 40*40 is 20*20. The average pooling computes the average

value within the 2*2 matrix, while the max pooling returns the maximum value of these four elements [103]. Max and average pooling are illustrated in the fig. (2.11) with a pooling size of 2*2.

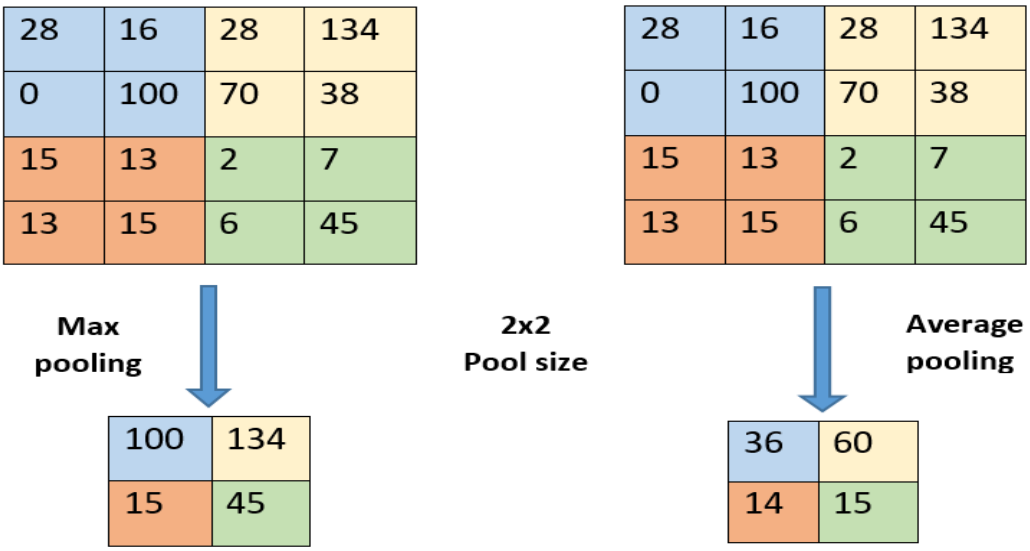


Fig. (2.11): Max and Average pooling with 2x2 pool size.

2.7.3 Fully connected layer

The fully connected layer is often used for classification at the network's termination [106]. It is comprised of a large number of neurons [107]. Each of them connects with all the neurons in the preceding and subsequent layers in a direct way, as shown in the fig. (2.12). The fully connected layer has a major drawback in that it requires complex computation in the training examples because it has a lot of parameters. This problem can be solved by the dropout technique, which removes some of the nodes and connections[105].

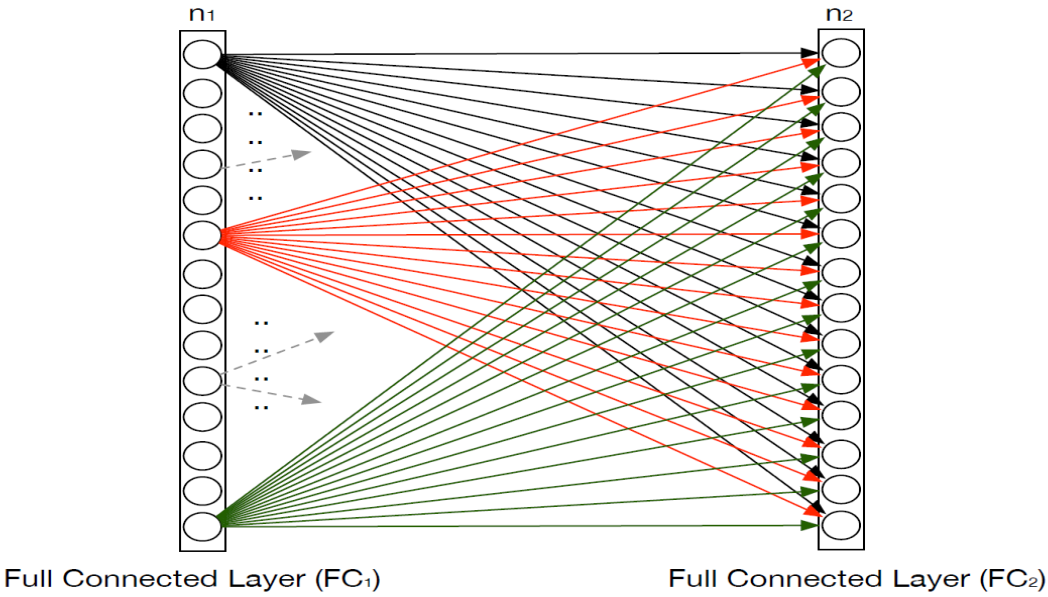


Fig. (2.12): Example of two fully connected layers [108].

2.7.4 Dropout and flatten layer

The dropout is applied to improve the performance of the neural networks by preventing overfitting problems. This is done by dropping out some units. These units are removed from the network randomly during the training phase [109] [110].

Flatten layers are used for reshaping the feature maps by converting the multidimensional output to a one-dimensional structure (vector). In CNN, the flatten layer is used to speed up the feedforward by reducing the parameters number and this is, in turn, to achieve higher accuracies[111] [112].

2.8 Types of CNN

There are several architectures for CNN, such as one-, two-, and three-dimension convolution convolutional neural networks (CNNs) and the first

two types are the most often used. The two-dimensional CNNs are applied to two-dimensional data types such as videos and images, while the one-dimensional CNNs, which are a modified version of two-dimensional CNNs, are applied to special types of data such as time series. Time series is a one-dimensional data type that is comprised of many signals with diverse scales [113] [114].

In dealing with 1D signals for specific applications, 1-D CNNs are more beneficial and hence superior to their 2-D counterparts because of the following reasons.

I. 1-D CNNs are easier to train than their equivalents 2-D CNNs.
II. 1-D CNNs don’t require special hardware setup such as GPU frame and cloud computing as 2-D CNNs.
III. Compact 1-D CNNs have minimal computational requirements that make it very well suited than 2-D CNNs in the applications of mini-cost and real-time, in particular on mobile devices [113].

2.9 DNNs evaluation

DNNs are evaluated using several measurements such as:

2.9.1 The evaluation metrics

The evaluation metrics are measurement tools which are used for measuring classifier performance. Deep neural networks (DNNs) and other classification algorithms have several popular metrics for evaluating classifier characteristics such as accuracy (Acc), sensitivity (Sen), specificity (Spec), precision (Pre), and F-measure. A learning method's classification might be a positive or negative class and four possible states can be classified under the correct or incorrect prediction [115][35]. The states are:

i. TP or true positive, it is the correct prediction of the positive class

ii. TN or true negative, it is the correct prediction of the negative class
iii. FP or false positive, it is the incorrect prediction of the positive class
iv. FN or false negative, it is the incorrect prediction of the negative class.

The accuracy is the most well-known evaluating metric which is used widely by a large number of researchers for evaluating trained classifiers and discriminating the best solution. It calculates the total instances that are predicted as correct when tested by a classifier with unseen data over the total instances [115], accuracy is calculated by equation (2.10):

$$Accuacy(Acc) = \frac{TP + TN}{TP + TN + FP + FN} \qquad (2.10)$$

The sensitivity or recall or true positive rate metric is used for measuring the number of samples that are predicted as positive to the total actually positive [116], and can be described by equation (2.11):

$$TPR = Sensitivity = Recall = \frac{TP}{TP + FN} \qquad (2.11)$$

The specificity metric is used for measuring the ratio of the samples that are predicted as negative to the total actually negative[115], as illustrated in equation (2.12):

$$Specificity(Spe) = \frac{TN}{TN + FP} \qquad (2.12)$$

Furthermore, a Precision metric, also known as positive predictivity, determines how many positively predicted samples are actually relevant [116], and it is calculated in equation (2.13) as:

$$Precision(Pre) = \frac{TP}{TP + FP} \qquad (2.13)$$

Finally, F-measure is the metric that is represented the harmonic mean between precision and recall values [115][116], F-measure is defined in equation (2.14) as:

$$F - measure = 2.\frac{precision \cdot recall}{precision + recall} \quad (2.14)$$

2.9.2 Receiver operating characteristic (ROC) curve

The receiver operating characteristic (ROC) curve is a measure of two-dimensional that is used to evaluate classification performance. ROC is using for comparing true to false positive rates. The Area under the ROC (AUC) curve is a popular scalar measure that gauges the performance of one facet [117][118] [119]. The location-based metrics ROC and AUC curves are illustrated in the fig. (2.13).

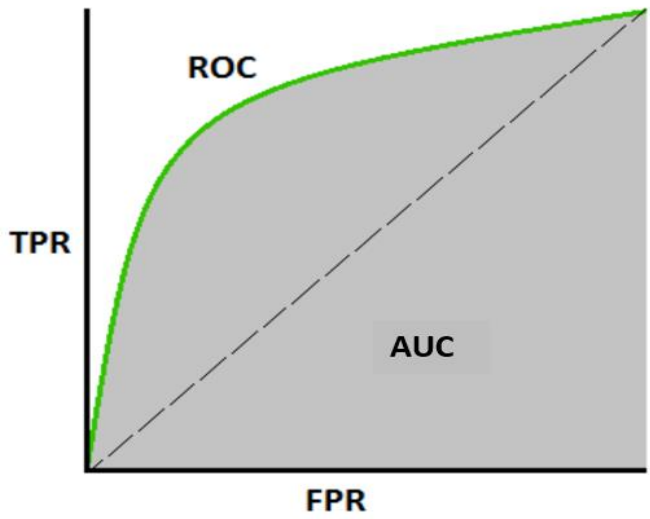


Fig. (2.13): ROC and AUC curves [120].

2.9.3 Confusion matrix

The confusion matrix (CM) is a commonly used tool to evaluate a model's performance and display its classification results in columns and rows as

shown in fig. (2.14) for each predicted and actual class, respectively. It is commonly used to discover classes that are frequently mispredicted or confused by one another [121][122]. The best confusion matrix contains a high number of instances down the major diagonal where the expected and actual classes are the same, and a small number of examples, or preferably none, in the other components where the predicted and actual classes are different [121].

		Actual Values	
Predicted Values		True Positive	False Positive
		False Negative	True Negative

Fig. (2.14): A typical Confusion Matrix

2.10 Normalization

Normalization is a scaling, mapping, or pre-processing method. At which a dataset can get a new range from an old one [123]. It is more than just a technique for converting original data into clean data, as well as improving the performance and helping in the prediction of a more accurate machine learning algorithm model [124]. The well-known normalization techniques are shown in the following subsections:

1) Z-score normalization

   Z-score normalization or statistical norm is a technique that normalizes each input feature vector by computing the mean and standard deviation and then employing them for each feature over a

set of training data [123]. The z-score normalization has shown in general equation (2.15) as:

$$x' = \frac{(x - \mu)}{\sigma} \quad (2.15)$$

Where x' is a normalized data , x is an original data, μ is the mean and σ is the standard deviation.

2) Min-max normalization

min-max normalization is applied to the dataset to change the numeric values into one range between 0 and 1 while keeping the differences between the values in the range. The min-max normalization has shown in equation (2.16) as:

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2.16)$$

Where z is the normalized data, x is the original data, and min (x) and max (x) are the minimum and maximum values respectively over a set of data.

Chapter 3 – Proposed Work

3.1 Introduction

In this chapter, the basic materials and methods implemented in the thesis work dealing with solving the research problem of arrhythmia detection and classification will be reviewed and discussed. Furthermore, the basic stages of the project are shown, namely the selected data set, pre-processing steps, which are include the dataset normalization, R-peaks detection, and the segmentation of the dataset into corresponding sequences, and splitting the dataset into training and testing dataset. In addition to feature extraction and classification will be illustrated. Finally, the parameters of the experiments will be explained and the proposed model hierarchy will be described.

3.2 Research method

The whole process in this model has been accomplished in basic four stages, the stages are:

1. Dataset gathering
2. Dataset preprocessing
3. Feature extraction
4. Learning/Classification

The full diagram of automatically arrhythmia classification is shown in the fig. (3.1).

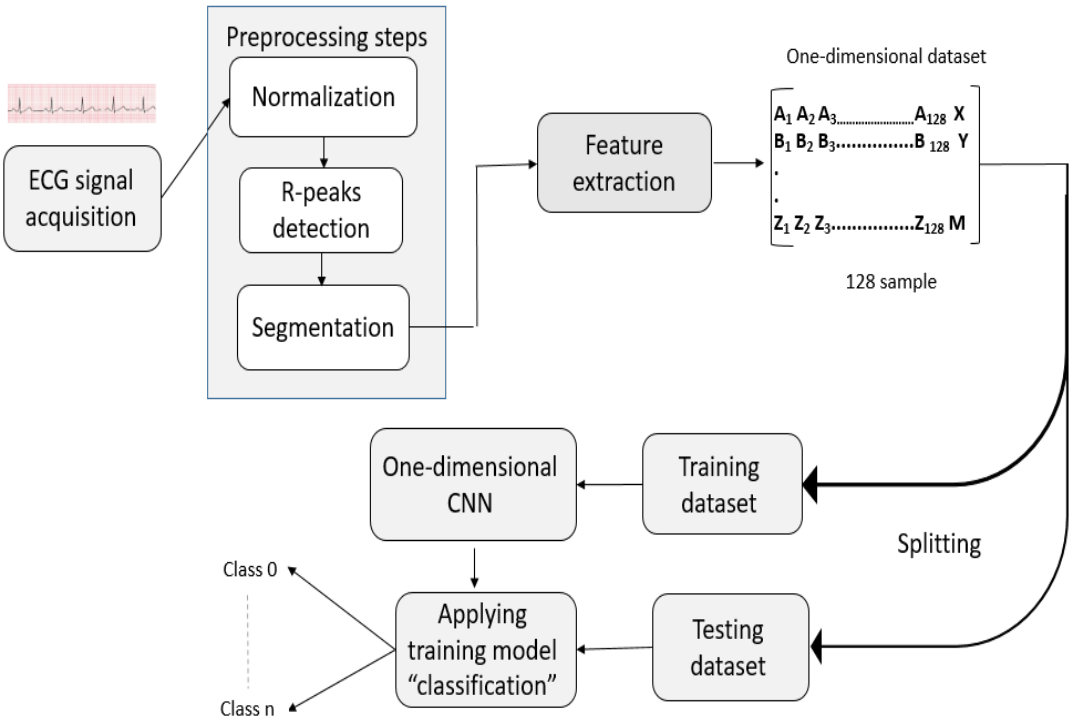


Fig. (3.1): Proposed Model Diagram

The whole system stages of the proposed system are explaining in algorithm 3.1 as follows:

Algorithm 3.1: The general proposed methodology system
Input: Raw ECG dataset
Output: CNN model for arrhythmia classification
1. collect of raw ECG dataset from MITDB
2. Normalization of raw ECG dataset using min-max normalization (call algorithm 3.2)
3. Detect R-peaks of normalized ECG dataset (call algorithm 3.3)
4. Segment normalized ECG dataset into sequences according to R-peaks (call algorithm 3.4) Cont.

5. Dividing sequences into training set and testing set
6. Construct proposed model 1-D CNN (call algorithm 3.5)
7.Evaluate proposed model 1-D CNN
7. End

3.3 Preprocessing steps

The dataset that selected for training and testing process has been preprocessed using normalization, R-peaks detection, and segmentation techniques, these techniques are presented in the following subsections:

3.3.1 Dataset normalization

In this step, the raw ECG signal from MITDB is very large values and therefore is incompatible with the network weights, so the values of the raw data are changed to a common scale without distorting the range values (or keeping the differences between the values in the range) and maintaining the signal properties. The min-max normalization method was applied in this thesis to change the numeric values into one range between 0 and 1. Table (3-1) shows some samples of raw and normalized ECG signals from MLII of record107.

Table (3-1): Sample of Raw and Normalized ECG Signal using Min-Max Normalization Process

No.	MLII	Normalization MLII
1	992	0.465339233038348
2	999	0.470501474926253
3	1013	0.480825958702064
4	1041	0.501474926253687
5	1089	0.536873156342182
6	1185	0.607669616519174
7	1304	0.695427728613569

8	1384	0.754424778761062
9	1402	0.767699115044247
10	1387	0.756637168141592
11	1370	0.74410029498525
12	1357	0.734513274336283
13	1348	0.72787610619469
14	1332	0.716076696165191
15	1312	0.701327433628318
16	1290	0.685103244837758
17	1276	0.674778761061946
18	1257	0.660766961651917
19	1244	0.651179941002949
20	1225	0.63716814159292

The normalization preprocessing step is explained in algorithm 3.2 as follows:

Algorithm 3.2: Normalization step
Input: Vector of ECG dataset (R), n:number of samples
Output: Vector of normalized dataset (S)
Begin
1. Find Min and Max values in the vector
2. For i =1 to n
3. modify any value i in the vector according to the equation (2.16)
4. End for

3.3.2 R-Peaks detection

After normalization of the dataset, the next step is R-peaks detection using the Scipy library with a threshold equal to 0.73. The R-peak has the highest amplitude among the rest of the waves of the ECG signal. This stage is useful for signify the beat and as a result, the ECG signals were segmented in to sequences. Fig. (3.2) illustrates the R-peaks detection for one minute of the ECG signal of record100.

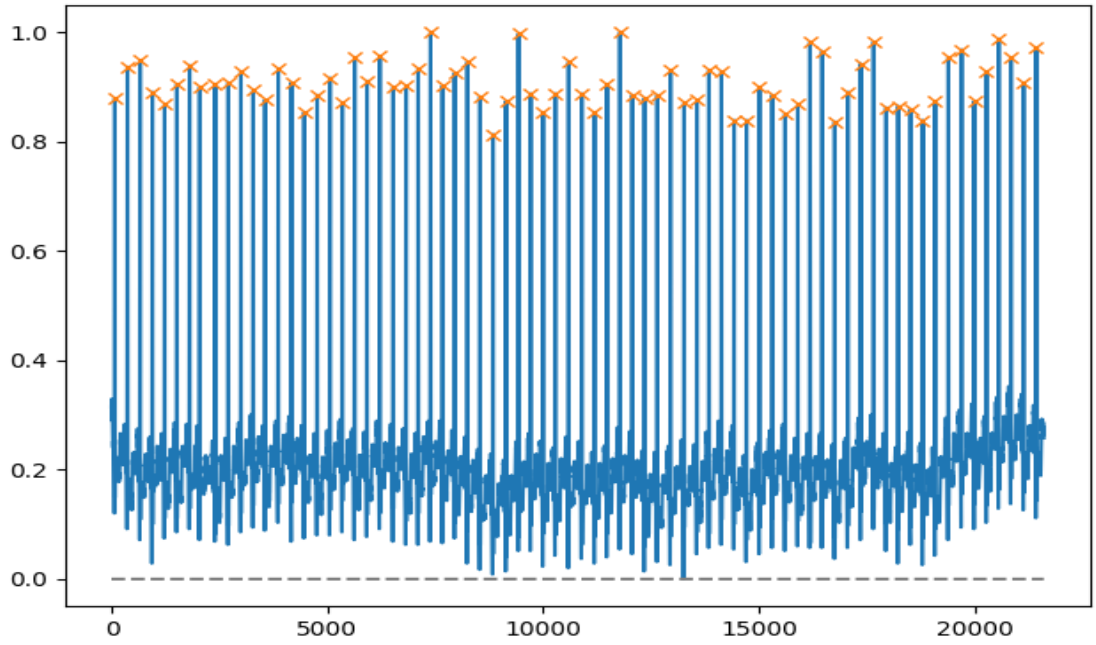


Fig. (3.2): R-peaks detection

The R-peaks detection step has been explained in algorithm 3.3 as follows:

Algorithm 3.3 : R-peaks detection step
Input: vector of normalized ECG dataset (NE_i), i :number of samples
Output: R-peaks of each recording, K :number of R-peaks of the normalized vector dataset, n: number of normalized samples in vector dataset
1. Let threshold = 0.73
2. Set K=0
3. For I =0 to n
4. Find R-peak at threshold using Scipy library
7. K=K+1
8. End for

3.3.3 Segmentation

The dataset which fed into our proposed model as form of sequences, each sequence includes one heartbeat, and these sequences are constituted through segmentation process.

In this thesis, the segmentation stage was accomplished by separating continued beat sequences into the relevant heartbeats segments. Each segment has only one ECG beat class, with 63 samples to the left of the initial R-peak and 64 samples to the right of the last identified unbroken R-peak to constitute a sequence of 128 samples, as shown in fig. (3.3). Also, the overlapping sequences have been removed.

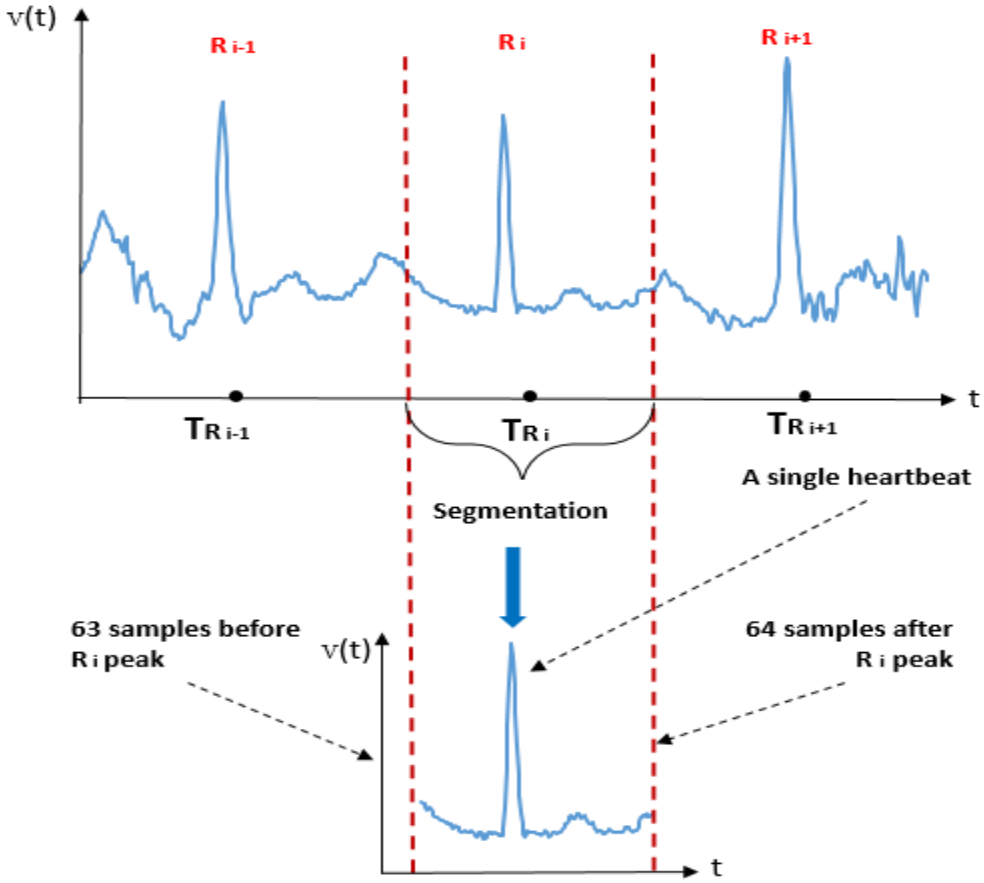


Fig. (3.3): Heartbeats segmentation process

A total of 43152 sequences were generated at the end of the dataset segmentation step, ready for the next phase. Segmentation stage has been explained in algorithm 3.4 as follows:

Algorithm 3.4 : Segmentation step
Input: Vector of normalized dataset , R-peaks, N :number of R-peaks
Output : N vectors of 128 samples
1.Let X is a vector of 63 samples before R-peaks and Y is a vector of 64 samples after R-peaks
2. let S is a vector of 128 samples
3.For i:1 to N
4.Set S[i] to null
5. S[i]= merge X with R-peak[i] and Y
6. End for

3.3.4 Training and testing dataset

After preprocessing steps have been completed, the dataset has been separated into a training set and a testing set. The training data set comprises the data that is used for system training, whereas the testing data set includes the unseen data class needed to evaluate the prediction performance of the model. In these experiments, we divided the dataset into 80% of the training set and 20% of the testing set. At each separation, the dataset was chosen randomly. The train/test divides were created in such a way that the two splits did not overlap. Fig. (3.4) depicts the proportion of separation.

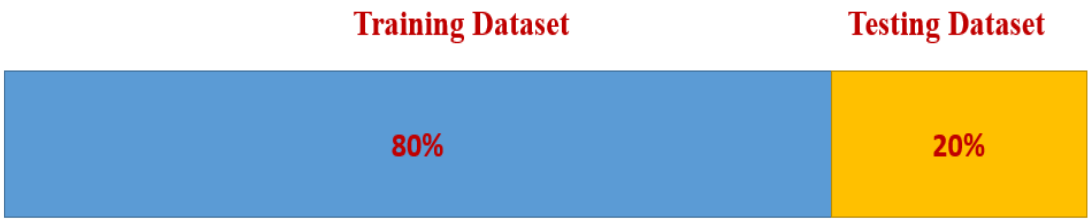


Fig. (3.4): The percentage of the training and testing set.

3.4 Feature extraction and classification

The two latest stages which are feature extraction and classification are accomplished by the one-dimensional convolutional neural network (1-D CNN). CNN which is automatic feature extractor has utilized to extract distinctive and robust arrhythmias features form ECG signal dataset. Then, these features are passed through the CNN's layers and finally the generated input vectors have been classified. The feature extraction and classification steps of the proposed model are illustrating in algorithm 3.5 as follows:

Algorithm 3.5: Implement the proposed 1-D CNN feature extraction and classification model's main steps
Input : ECG sequences Is = {I1,I2,I3,I4,..,In}, n : number of sequences, Epoch : number of epochs.
Output: 1-D CNN
Training step:
1. Split Is into two parts (training $_{set}$, testing $_{set}$)
2. I $_{train}$ ←Train (1-D CNN, training $_{set}$)
2.1 Initialize the weights at random values.
2.2 Take the input from training data and pass it through the CNN layers.
2.3 Calculate the output's error according the equation (2.1)
Cont.

2.4 Update the weights according to Adam optimizer.
2.5 Perform steps 2.2-2.4 to overall training $_{set}$ for each epoch
2.6 Repeat steps 2.2-2.5 until the error decreased to acceptable tolerance.
3. Finish training when reached acceptable performance.
Testing step:
4. Use the trained model I_{train} for predict the testing $_{set}$ label.
5. Get the known labels for testing $_{set}$
6. Evaluation 1-D CNN by using metrics (mean accuracy, sensitivity, specificity, precision, and f1-score) as equation (2.10), (2.11), (2.12), (2.13), (2.14) respectively.
7. End

3.5 Experimentations' parameters

The experiments' main goal was to see if a convolutional neural network (CNN) could classify arrhythmias using ECG signal data with a high degree of accuracy. A one-dimensional convolutional neural network (1-D CNN) model was presented for experiments to address this issue. Starting with typical or default values, the influence of different hyper parameters was investigated by performing experiments with modifications to a single parameter every time.

The hyper parameter tuning is very necessary and it is done manually every time randomly. For the proposed model, the default learning rate is used, which is equal to 0.001 without adjustment, whereas the number of epochs ranges from 5 to 50. Furthermore, the batch size is set to 20. The

training and testing sets are split as illustrated above, with training and testing sizes equal to 0.80 and 0.20, respectively. In addition, the optimizer that has been applied in the proposed model is the Adam optimizer as well as the cost function is sparse-categorical cross entropy. Finally, the activation function which is used for most layers is the ReLU function, except for the last dense layer, which utilizes the softmax function to get the final classes.

3.2 Model architecture

The proposed model is based on one-dimensional CNN (1-D CNN). It has been applied multiple, once for the binary classification and the other for the multi-class (three-, four-, six- and seven-class) classification. This model consists of nine layers distributed as follow:

- Three 1D convolution layers
- One maxpooling 1D layer
- One dropout layer
- One Flatten layer
- Three Fully connected layers(dense)

In the proposed model, the network hierarchy shares the same parameters for all different schemes of classification except for the last layer (fully-connected layer 3) which is 2050 for binary classification and 3075, 4100, 6150, and 7175 for three-, four-, six-class, and seven-class classification, respectively.

Tables (3-2) and (3-3) show the summary of the proposed model for seven-class and binary classification, respectively. The architecture of the proposed nine-layer CNN model for seven-class classification is illustrated in Fig. (3.5).

Table (3-2): The hierarchy details of the proposed CNN model for seven-class classification

Layer	*Type*	*Filter size*	*Stride*	*Kernel*	*Activation function*	*Parameters*
1	Conv.1D	5	1	32	ReLU	192
2	Conv.1D	5	1	64	ReLU	10304
3	Conv.1D	5	1	128	ReLU	41088
4	Maxpooling1D	5	2	-	-	-
5	Dropout	0.5	-	-	-	-
6	Flatten	-	-	-	-	-
7	Fully connected	-	-	512	ReLU	4194816
8	Fully connected	-	-	1024	ReLU	525312
9	Fully connected	-	-	7	Softmax	7175

Table (3-3): The hierarchy details of the proposed CNN model for binary classification.

Layer	*Type*	*Filter size*	*Stride*	*Kernel*	*Activation function*	*Parameters*
1	Conv.1D	5	1	32	ReLU	192
2	Conv.1D	5	1	64	ReLU	10304
3	Conv.1D	5	1	128	ReLU	41088
4	Maxpooling1D	5	2	-	-	-
5	Dropout	0.5	-	-	-	-
6	Flatten	-	-	-	-	-
7	Fully connected	-	-	512	ReLU	4194816
8	Fully connected	-	-	1024	ReLU	525312
9	Fully connected	-	-	2	Softmax	2050

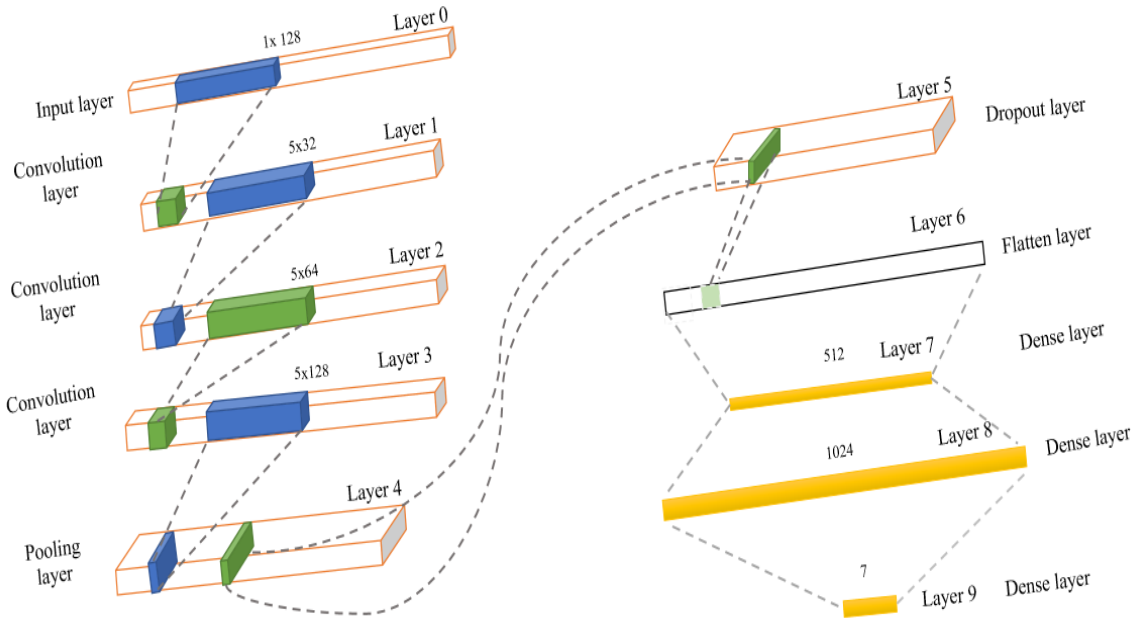


Fig. (3.5): The architecture of the proposed 1-D CNN model

Chapter 4 – Experiments and Results

4.1 Overview

In this chapter, the results obtained from the proposed model for different experiments will be displayed and discussed. Tensor flow, a platform for building and developing machine learning models, was used to design, train, and test the proposed model. Using the open source Keras library, which provides many degrees of abstraction for building and training models with large-scale layers.

We evaluated binary, three, four, six, and seven classes of arrhythmia with the proposed 1-D CNN model with different experiments. Due to the large size of the normal class relative to the other categories in the binary and multi-category classification, different sizes have been utilized in the classification of four, six, and seven categories. As for the classification of the three categories, the normal category was not used, instead we adopted the use only three types of arrhythmia, which are close in the amount of data.

On the other hand, the epoch number was changed in each experiment in order to test its influence on the training and performance of the model. The epoch number ranged from 5 to 50 (with an increase of 5 at each time). While, the batch size is 20 for all experiments. Five metrics have been used for the evaluation of the proposed model, which are accuracy, precision, recall, f1-score, and specificity, in addition to the confusion matrix and ROC-curve. Also, the training time for each experiment has recorded in seconds.

4.2 ECG dataset

For this thesis, ECG arrhythmia datasets are obtained from the open-source MIT-BIH Arrhythmia database (MITDB), which provides the ECG signal dataset that comprises 48 half-hour recordings from 47 subjects reported by

the BIH arrhythmia laboratory between 1975-1979. The first twenty-three recordings numbered 100-124. The remaining twenty-five recordings were chosen from the same set, with labels ranging from 200 to 234 [10]. These subjects included 22 women with an age from 23 to 89 years old and 25 men with an age from 32 to 89 years. These records are obtained from an ECG Holter recording device with MLII and V1-V5 leads which are varying among subjects. A companion file was attached to each recording known as the reference annotation file that described each beat, where two cardiologists annotated each record independently. In all recordings, there are approximately 100,000 annotations [10].

According to the contents of the records, 31 records were selected from a total of 48 records from a single lead which is a modified limb lead (MLII), which are contained the relevant arrhythmias such as normal (N), left bundle branch block (LBBB), right bundle branch block (RBBB), atrial premature beat (APB), premature ventricular contraction (PVC), fusion beat (F), and paced beat (p) as shown in fig. (4.1).

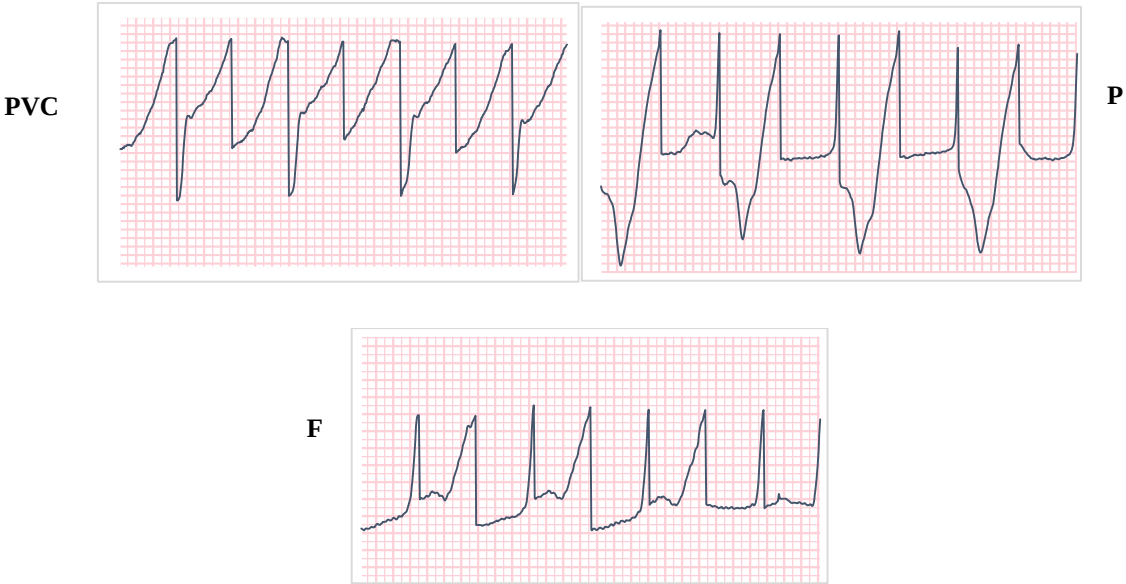


Fig. (4.1): ECG waveforms of seven heartbeat classes

4.3 Details of experiments' datasets

Different experiments were applied in this thesis with different groups of dataset. as illustrated in the following subsections:

4.3.1 Binary experiment

In binary classification, 43,152 sequences comprising normal and abnormal beats (all but normal beat classes considered to be abnormal) were used. Then the original normal dataset is reduced to half and used with the same type of abnormal class. Furthermore, the normal dataset is reduced more into 8000 and 6000 sequences for using with the same abnormal dataset in separated experiments. A summary of the binary classes with full, half, 8000 and 6000 normal datasets is shown in table (4-1).

Table (4-1): The Summary of the binary classes with overall, half, 8000, and 6000 normal datasets.

Class / Experiment	Normal	Percent	Abnormal	Percent	Total
1	30777	71.3%	12375	28.7%	43152
2	15000	45.8%	12375	45.2%	27375
3	8000	39.3%	12375	60.7%	20375
4	6000	32.7%	12375	67.3%	18375

For each scheme of binary classification, the summary of training and testing datasets is explained in table (4-2).

Table (4-2): The summary of training and testing dataset for binary experiment

Normal dataset size	Training set	Testing set	Total
30777	34521	8631	43152
15000	21900	5475	27375
8000	16300	4075	20375
6000	14700	3675	18375

4.3.2 Three-class experiment

Three-class classification is multi-class classification with three different types of classes. In this thesis, three various types of arrhythmia are classified which are left bundle branch block (LBBB), right bundle branch block (RBBB), and atrial premature beat (APB) with training dataset of 6651 and

testing dataset of 1663. The summary of three-class classification presented in table (4-3).

Table (4-3): The summary of dataset for three classes experiment

Class	LBBB	RBBB	APB	Total
Dataset	3798	3144	1372	8314
Percent	45.7%	37.8%	16.5%	

4.3.3 Four-class experiment

In four-class classification, four different ECG beats were classified which are N, LBBB, RBBB, and APB.

Various normal dataset size was implemented in the experiments, in four-class classification with full normal dataset, 30777 normal sequences are employed in this classification with the three various arrhythmias beats (LBBB, RBBB, and APB). Then, half normal dataset (15000 normal sequences) are used with the same types of arrhythmia. Normal datasets are reduced more to 8000 and 6000 sequences and utilized with the same arrhythmia's types in different experiments, the summary of four-class classification are illustrated in table (4-4).

Table (4-4): The summary of dataset for four classes experiment

Class	N	LBBB	RBBB	APB	Total
Dataset	30777	3798	3144	1372	39291
	15000	3798	3144	1372	23314
	8000	3798	3144	1372	16314
	6000	3798	3144	1372	14314

The percentage of four classes arrhythmia are shown in table (4-5).

Table (4-5): The percentage of four classes dataset of arrhythmia

Normal dataset	Class / Percent	N	LBBB	RBBB	APB
30777		78.7%	9.7%	8.0%	3.5%
15000		64.3%	16.3%	13.5%	5.9%
8000		49.0%	23.3%	19.3%	8.4%
6000		41.9%	26.5%	22.0%	9.6%

For each scheme of four-class classification, the summary of training and testing datasets is explained in table (4-6).

Table (4-6): The summary of training and testing dataset for four classes experiment

Normal dataset size	Training set	Testing set	Total
30777	31272	7819	39291
15000	18651	4663	23314
8000	13051	3263	16314
6000	11451	2863	14314

4.3.4 Six-class experiment

In six-class classification, six classes of ECG signals which are N, LBBB, RBBB, APB, PVC, and F with different experiments according to normal dataset and epochs number. Firstly, full normal dataset is used with the remainder arrhythmias classes, then the amount of normal dataset has reduced into the half and used in the classification with the remainder arrhythmias classes. Also, normal dataset has reduced to 8000 and 6000

sequences and used with the remainder five classes at each time. The summary and the percentage of six classes dataset with full, half, 8000, and 6000 normal datasets were illustrated in the table (4-7) and table (4-8), respectively. The summary of training and testing datasets is explained in table (4-9).

Table (4-7): The summary of dataset for six classes experiment

Class	N	LBBB	RBBB	APB	PVC	F	Total
Dataset	30777	3798	3144	1372	1342	368	40801
	15000	3798	3144	1372	1342	368	25024
	8000	3798	3144	1372	1342	368	18024
	6000	3798	3144	1372	1342	368	16024

Table (4-8): The percentage of six classes dataset of arrhythmia

Normal Data	Class / Percent	N	LBBB	RBBB	APB	PVC	F
30777		75.4%	9.3%	7.7%	3.4%	3.3%	0.9%
15000		59.9%	15.2%	12.6%	5.5%	5.4%	1.5%
8000		44.4%	21.1%	17.4%	7.6%	7.4%	2.0%
6000		37.%	23.7%	19.6%	8.6%	8.4%	2.3%

Table (4-9): The summary of training and testing dataset for six-class experiment

Normal dataset size	Training set	Testing set	Total
30777	32640	8161	40801
15000	20019	5005	25024
8000	14419	3605	18024
6000	12819	3205	16024

4.3.5 Seven-class experiment

Seven-class classification has implemented in the last experiments that include seven classes of ECG signals which are N, LBBB, RBBB, P, APB, PVC, and F with different normal dataset size. Firstly, full normal dataset with six types of arrhythmias are used for classification. Then, reduced the normal dataset to 15000, 8000, and 6000 and applied with the same set of six classes of arrhythmia in a set of experiments. The table (4-10) presented the summary of seven-class classification at each amount of normal dataset.

Table (4-10): The Summary of dataset for seven-class experiment

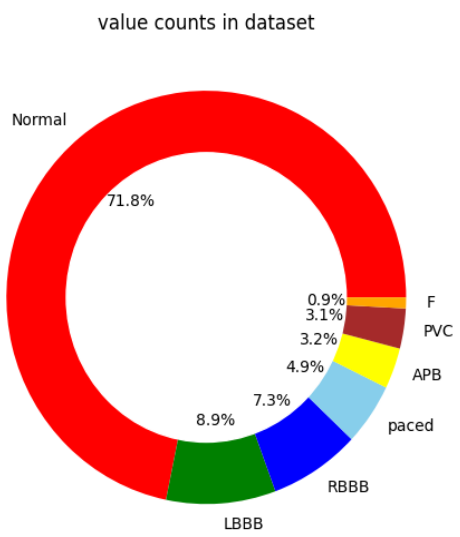
Class	N	LBBB	RBBB	P	APB	PVC	F	Total
Dataset	30777	3798	3144	2083	1372	1342	368	42884
	15000	=	=	=	=	=	=	27107
	8000	=	=	=	=	=	=	20107
	6000	=	=	=	=	=	=	18107

The summary of the training and testing dataset for the seven-class classification is shown in the table (4-11).

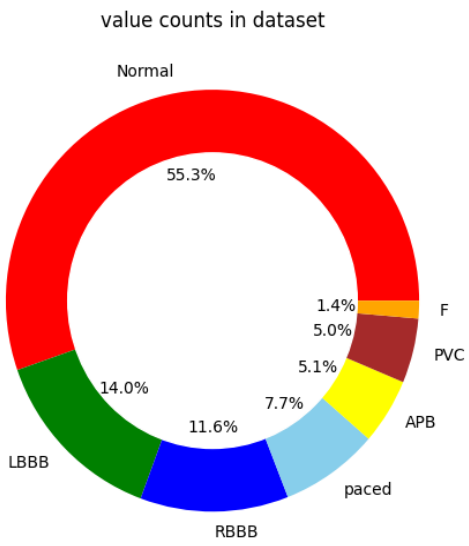
Table (4-11): The summary of training and testing set for seven classes experiment

Normal dataset size	Training set	Testing set	Total
30777	34307	8577	42884
15000	21685	5422	27107
8000	16085	4022	20107
6000	14485	3622	18107

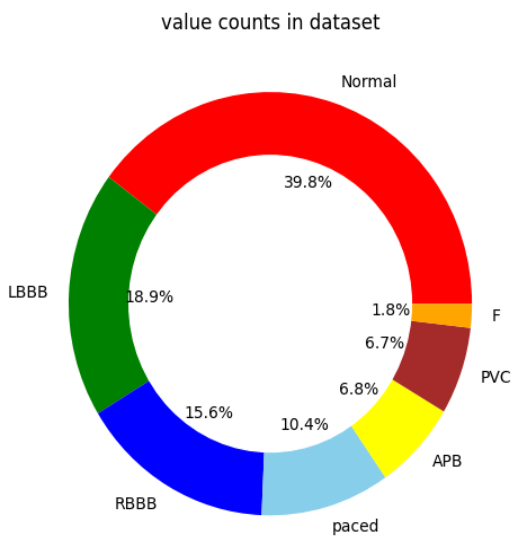
The percentages of seven classes ECG signals for each amount normal dataset are shown in the fig. (4.2) (a) (b) (c) (d).



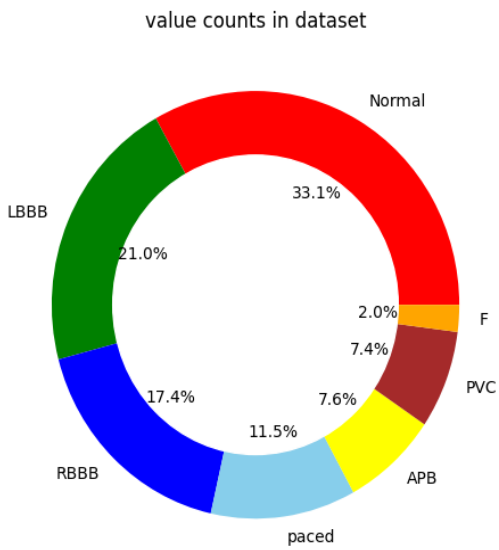
(a) Full normal dataset



(b) Half normal dataset



(c) 8000 normal dataset



(d) 6000 normal dataset

Fig. (4.2): The proportion of seven classes of arrhythmia dataset.

4.4 Experimental setup

The experiments were carried out on a personal computer with an intel® Core™ i7- 6820HQ central processing unit(CPU) 2.70 GHz, 16 GB RAM, 64-bit operating system, x64-based processor. no additional hardware device is required such as graphical processing unit (GPU). The application for

performing the experiments was run by Python 3.8 on PyCharm, which is a code editor dedicated to the Python programming language for processing ECG signals. Keras based on the open-source TensorFlow framework used for creating DNN. Other libraries were used in this thesis, such as the SciPy library for signal preprocessing and pandas, NumPy, and the Sklearn library for dataset manipulation and analysis.

4.5 The experimented results

As mentioned, there are different experiments that were applied in practical work with the proposed model, so we will describe them in detail in the following sections according to the number of classes.

4.5.1 Binary classification

In binary classification, the data set includes only two categories, which are normal and abnormal categories. In other words, the proposed model detects whether the ECG signal is normal or not. In this thesis, several experiments with binary classification were carried out depending on the size of the normal data set.

Firstly, the overall normal dataset is used in binary classification. We presented the performance of the proposed model in these experiments by the evaluation metrics with varied epoch sizes in each experiment.

The results of these experiments are illustrated in table (4-12). Then We searched for the experiment with the best results among all. The confusion matrix of the highest results of CNN model of binary classification with overall normal dataset of the experiment at epoch 45 has been shown in fig. (4.3).

Table (4-12): The performance of the binary classification with full normal dataset						
Metric / Epoch no.	Acc%	Pre%	Sen %	F1%	Spec%	Running Time in sec.
5	0.9847	0.9853	0.9779	0.9816	0.9779	209.4
10	0.9864	0.9865	0.9808	0.9837	0.9808	455.2
15	0.9893	0.9904	0.9839	0.9872	0.9839	662.5
20	0.9910	0.9909	0.9873	0.9891	0.9873	938.9
25	0.9888	0.9894	0.9835	0.9865	0.9835	1158.8
30	0.9908	0.9919	0.9861	0.9890	0.9861	1322.5
35	0.9903	0.9907	0.9858	0.9883	0.9858	1471.3
40	0.9892	0.9899	0.9842	0.9870	0.9842	1756.1
45	0.9927	0.9926	0.9898	0.9912	0.9898	1896.9
50	0.9900	0.9891	0.9869	0.9880	0.9869	2157.6

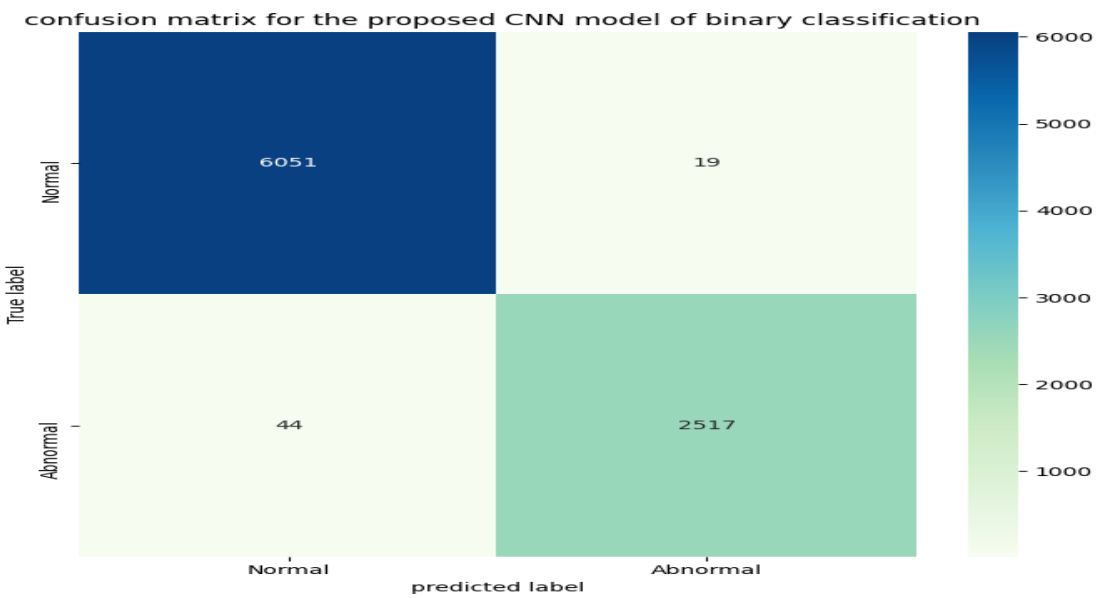


Fig. (4.3): Confusion matrix of the CNN model for binary classification with full normal dataset.

In binary classification with 15000 a normal dataset, we reduced the normal dataset by choosing randomly approximately half of a normal dataset. The

performance of the proposed model with these datasets has been shown in table (4-13). The confusion matrix of the experiment at epoch 45 (the highest results) has been shown in fig. (4.4).

Table (4-13): The performance of the binary classification with 15000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Running Time in sec.
5	0.9748	0.9753	0.9737	0.9745	0.9737	173.7
10	0.9775	0.9796	0.9753	0.9775	0.9753	287.2
15	0.9870	0.9875	0.9862	0.9869	0.9862	428.2
20	0.9865	0.9871	0.9856	0.9863	0.9856	520.4
25	0.9887	0.9887	0.9884	0.9885	0.9884	647.0
30	0.9867	0.9873	0.9857	0.9865	0.9857	789.1
35	0.9883	0.9889	0.9874	0.9882	0.9874	962.8
40	0.9887	0.9887	0.9883	0.9885	0.9883	1099.9
45	0.9909	0.9904	0.9910	0.9907	0.9905	1197.5
50	0.9850	0.9862	0.9836	0.9849	0.9836	1441.5

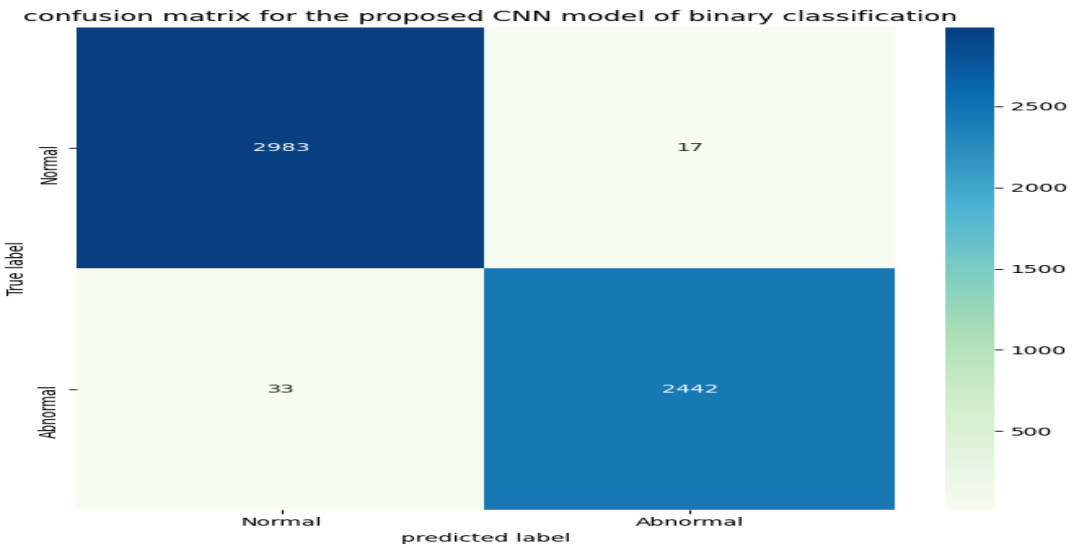


Fig. (4.4): Confusion matrix of the proposed CNN model for binary classification with half normal dataset.

In binary classification with a normal dataset equal to 8000 sequences, the performance is achieved with ten experiments with varied epoch numbers are presented in table (4-14). The confusion matrix of the proposed model for the highest results experiment at epoch 50 is represented in fig. (4.5).

Table (4-14): The performance of the binary classification with 8000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Running time in sec.
5	0.9816	0.9812	0.9799	0.9806	0.9799	102.4
10	0.9845	0.9832	0.9842	0.9837	0.9842	207.5
15	0.9841	0.9828	0.9836	0.9832	0.9836	299.5
20	0.9858	0.9838	0.9865	0.9851	0.9865	552.0
25	0.9858	0.9855	0.9844	0.9850	0.9844	510.9
30	0.9872	0.9864	0.9868	0.9866	0.9868	742.9
35	0.9890	0.9878	0.9890	0.9884	0.9890	693.6
40	0.9870	0.9852	0.9876	0.9864	0.9876	765.8
45	0.9850	0.9829	0.9859	0.9844	0.9859	877.7
50	0.9850	0.9829	0.9859	0.9844	0.9859	1084.6

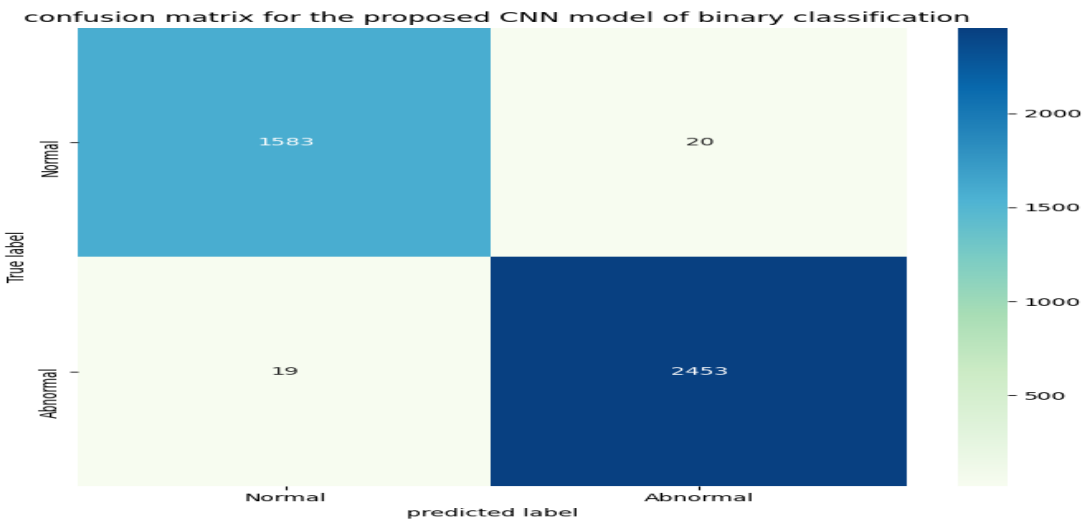


Fig. (4.5): Confusion matrix of the proposed CNN model for binary classification with 8000 normal datasets.

In binary classification with 6000 normal datasets, the performances have been achieved are presented in the table (4-15). The confusion matrix of the proposed model with the highest performance of the binary classes with 6000 normal datasets has shown in fig. (4.6).

Table (4-15): The Performance of the binary Classification with 6000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Running time in sec.
5	0.9840	0.9839	0.9795	0.9817	0.9795	112.9
10	0.9864	0.9829	0.9862	0.9846	0.9862	250.4
15	0.9880	0.9858	0.9870	0.9864	0.9870	268.4
20	0.9867	0.9832	0.9867	0.9849	0.9867	362.5
25	0.9894	0.9866	0.9893	0.9880	0.9893	462.0
30	0.9899	0.9872	0.9899	0.9886	0.9899	550.4
35	0.9886	0.9849	0.9894	0.9871	0.9894	655.0
40	0.9918	0.9907	0.9907	0.9907	0.9907	737.7
45	0.9891	0.9866	0.9887	0.9877	0.9887	836.4
50	0.9921	0.9913	0.9906	0.9910	0.9906	954.1

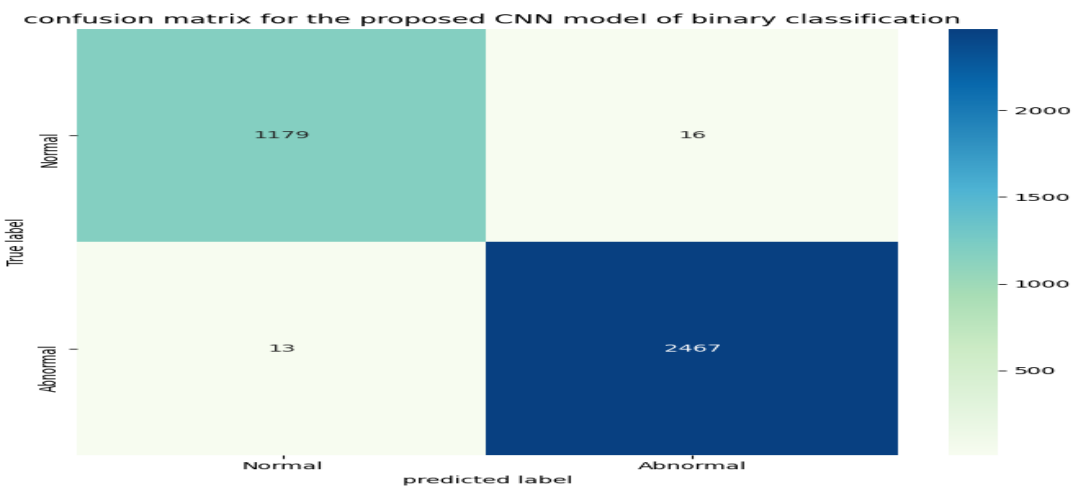


Fig. (4.6): Confusion matrix of the proposed CNN model for binary classification with 6000 normal datasets.

The ROC-curves of the proposed model for the binary classification with full, half, 8000, and 6000 normal datasets have been represented in fig. (4.7) (a) (b) (c) (d), respectively.

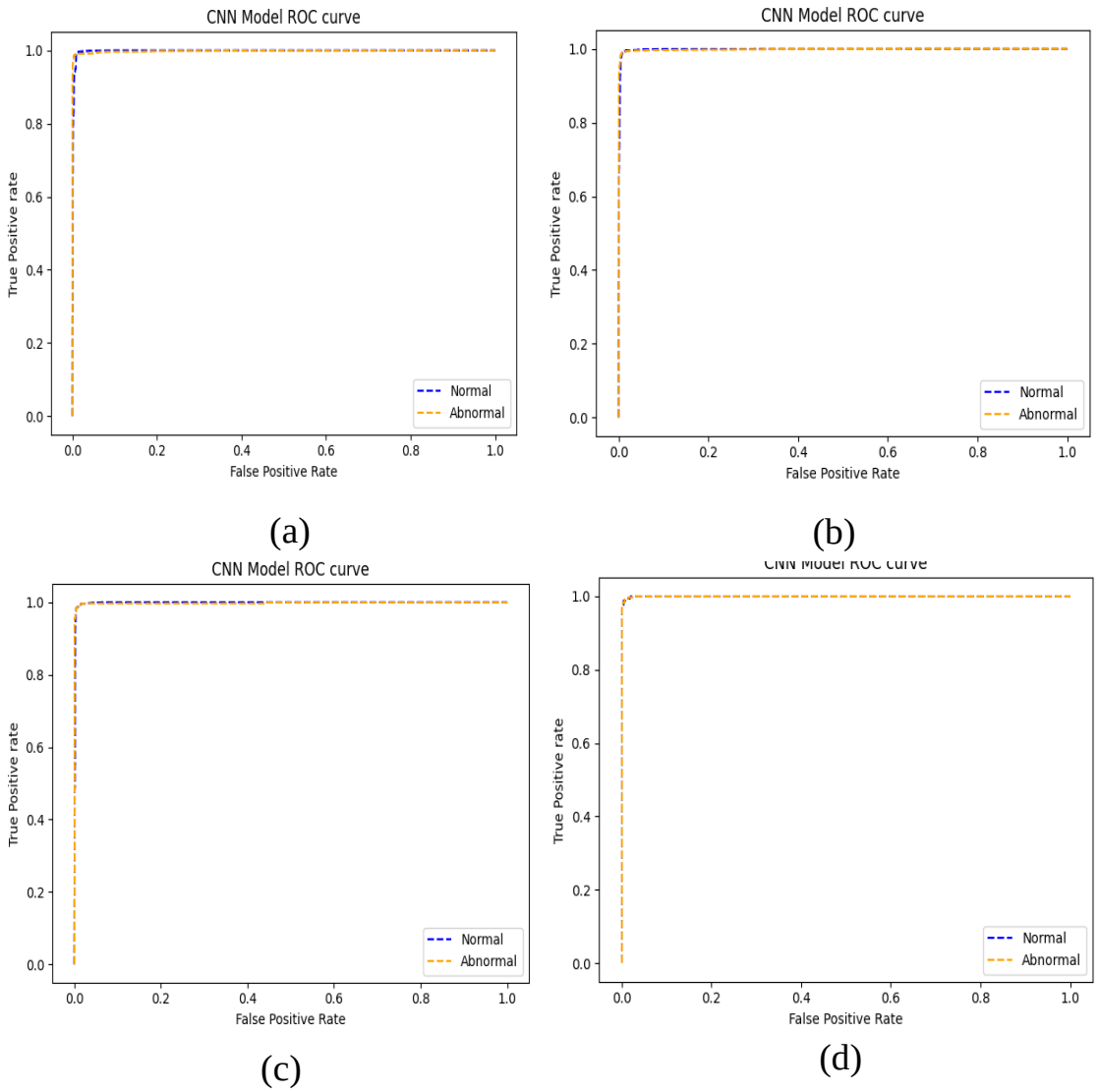
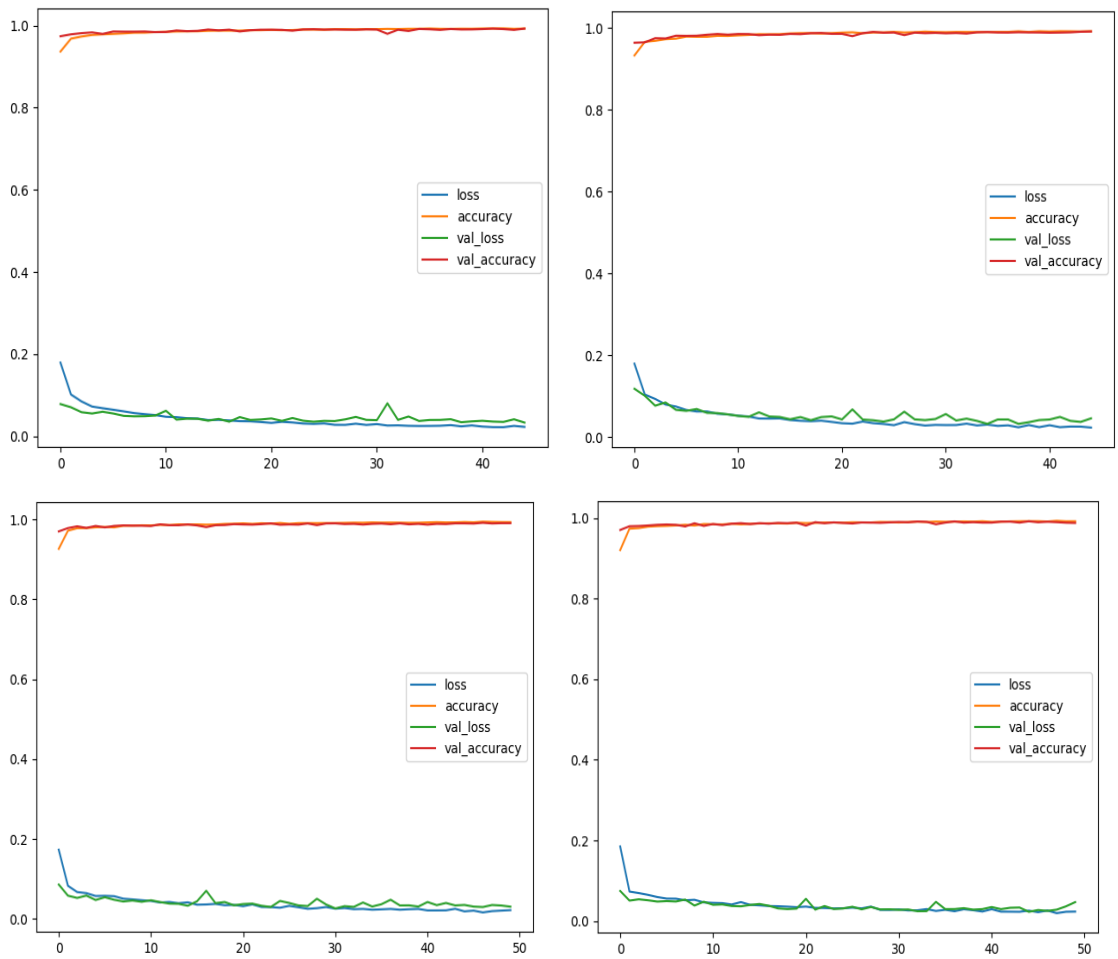


Fig. (4.7): ROC-curves of proposed model for binary classification with (a)full normal datasets, (b) half normal dataset, (c) 8000 normal dataset (d) 6000 normal dataset

The loss, accuracy, validation-loss, and validation-accuracy of the proposed model for the binary classification with full, half, 8000, and 6000 normal datasets have been represented in fig. (4.8) (e) (f) (g) (h), respectively.

(e) (f)



(g) (h)

Fig. (4.8): Loss, accuracy, validation loss, and validation accuracy of proposed model for binary classification with (e)full normal datasets, (f) half normal dataset, (g) 8000 normal dataset (h) 6000 normal dataset.

4.5.2 Three-class classification

In three class classification, three different classes of arrhythmia have been diagnosed by the proposed algorithm named LBBB, RBBB, and APB. Ten experiments with varied epoch number at each experiment has implemented. The performance of three class classification have been shown in table (4-16).

Table (4-16): The performance of three class classification						
Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9675	0.9724	0.9305	0.9510	0.9824	41.3
10	0.9784	0.9684	0.9680	0.9682	0.9892	95.9
15	0.9850	0.9807	0.9757	0.9782	0.9922	138.1
20	0.9910	0.9898	0.9842	0.9870	0.9952	186.5
25	0.9922	0.9902	0.9866	0.9884	0.9959	208.7
30	0.9940	0.9923	0.9899	0.9911	0.9969	361.6
35	0.9952	0.9934	0.9926	0.9930	0.9976	361.4
40	0.9952	0.9942	0.9917	0.9929	0.9975	375.6
45	0.9946	0.9913	0.9929	0.9921	0.9974	403.9
50	0.9946	0.9937	0.9904	0.9920	0.9972	468.1

Confusion matrices of the proposed model for the three-class classification of experiment at epoch 35 are shown in fig. (4.9).

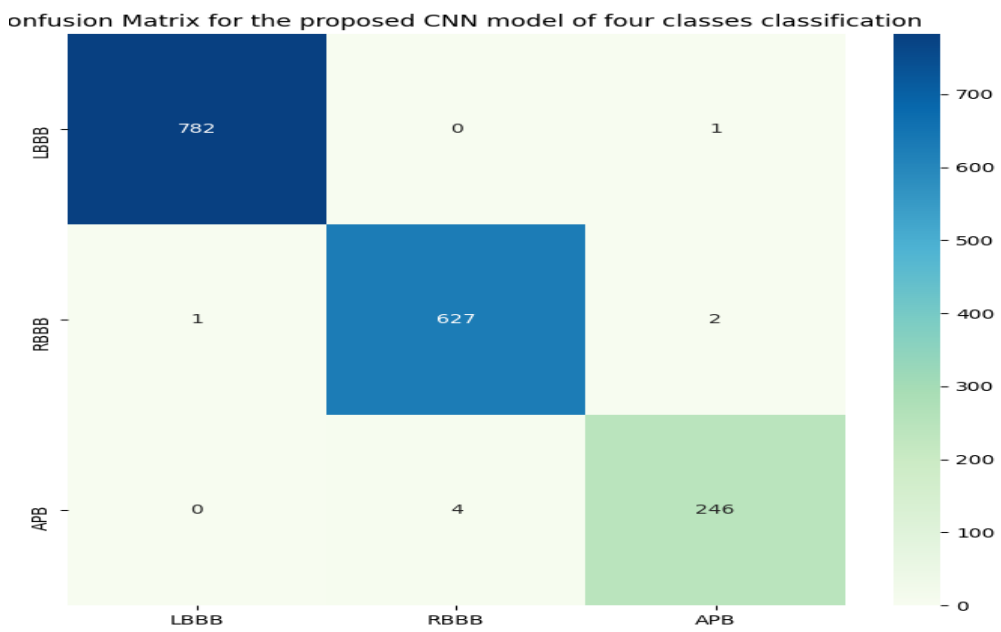


Fig. (4.9): Confusion matrix of the proposed CNN model for three classes of arrhythmia

The ROC-curves and the loss, accuracy, validation-loss, and validation-accuracy of the proposed model for three-class classification have been represented in fig. (4.10).

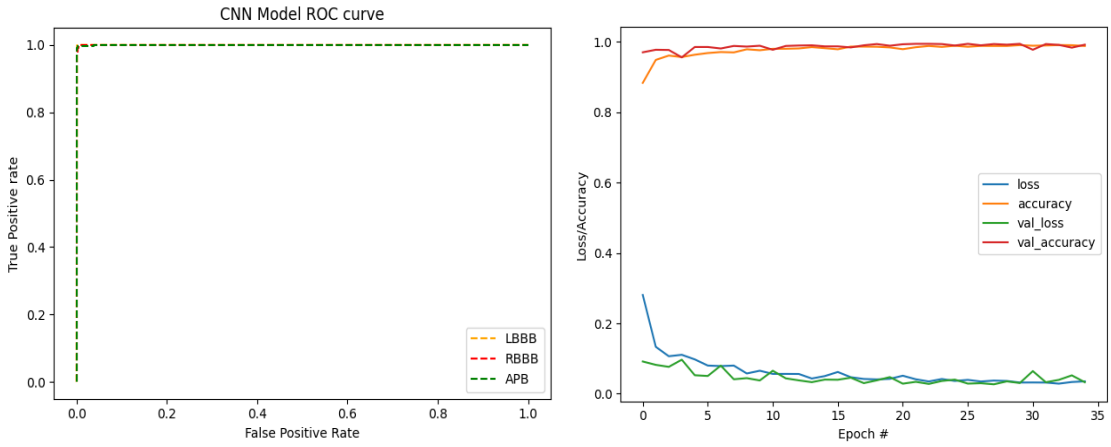


Fig. (4.10): Left: ROC-curve, right: loss, accuracy, validation loss, and validation accuracy of proposed model for three classes of arrhythmia dataset.

4.5.3 Four-class classification

In four-class classification, four different ECG beats were classified which are N (normal), LBBB (left bundle branch block), RBBB (right bundle branch block), and APB (atrial premature beat).

Various normal dataset size has been implemented in the experiments as shown below. The first experiments in the classification with four-class were implemented with overall normal dataset. The performance of four-class classification with overall normal dataset has shown in table (4-17).

Table (4-17): The performance of four-class classification with full normal dataset						
Metric / Epoch no.	Acc %	Pre%	Sen%	F1%	Spec%	Training time in sec.
5	0.9870	0.9662	0.9369	0.9513	0.9893	332.9
10	0.9845	0.9790	0.8979	0.9367	0.9836	806.8
15	0.9909	0.9808	0.9478	0.9640	0.9912	941.0
20	0.9909	0.9722	0.9572	0.9647	0.9923	911.0
25	0.9932	0.9808	0.9641	0.9724	0.9943	1115.9
30	0.9930	0.9816	0.9610	0.9712	0.9937	1186.5
35	0.9881	0.9515	0.9428	0.9472	0.9919	1745.1
40	0.9916	0.9727	0.9621	0.9674	0.9931	3340.6
45	0.9931	0.9759	0.9694	0.9726	0.9948	1855.8
50	0.9932	0.9834	0.9622	0.9727	0.9938	2188.6

The confusion matrix for the four-class classification for overall normal dataset with the experiment at epoch 45 is shown in fig. (4.11).

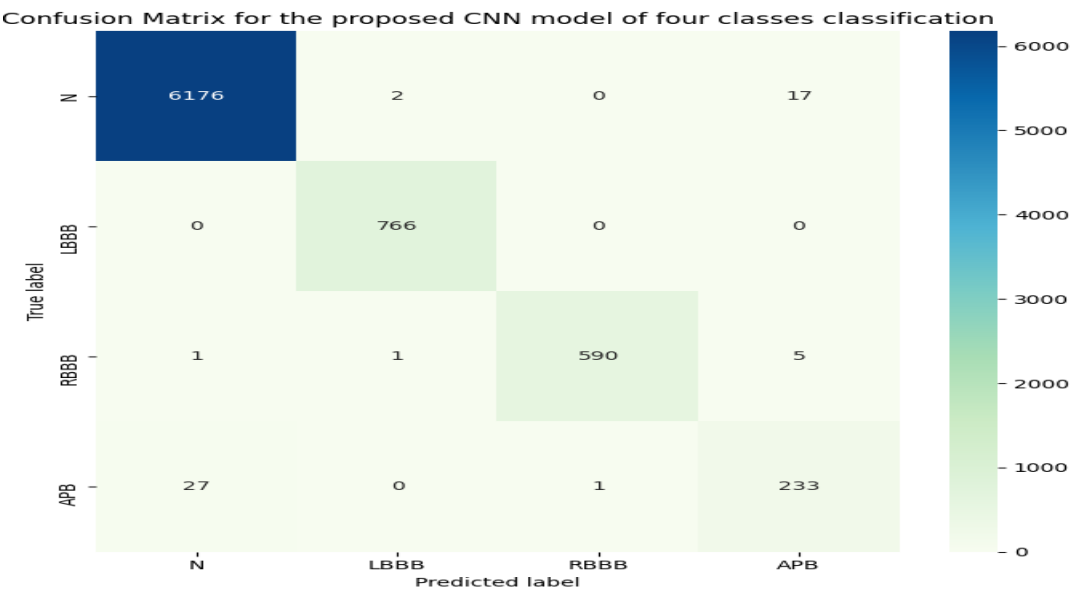


Fig. (4.11): Confusion matrix of the proposed CNN model for four-class classification with overall normal datasets.

The performance of experiments of four-class classification with half normal dataset (15000 sequences normal dataset) has shown in table (4-18). The confusion matrix of the proposed model with experiment at epoch 40 (highest results) has been illustrated in fig. (4.12).

Table (4-18): The performance of four-class classification with 15000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1%	Spec %	Training time in sec.
5	0.9749	0.9577	0.9749	0.9383	0.9860	117.9
10	0.9790	0.9645	0.9308	0.9474	0.9883	232.1
15	0.9837	0.9821	0.9399	0.9606	0.9899	350.4
20	0.9831	0.9851	0.9377	0.9608	0.9892	470.4
25	0.9843	0.9860	0.9411	0.9630	0.9902	587.1
30	0.9903	0.9840	0.9747	0.9793	0.9947	962.6
35	0.9824	0.9683	0.9520	0.9601	0.9906	1186.6
40	0.9899	0.9884	0.9652	0.9767	0.9939	934.2
45	0.9897	0.9903	0.9624	0.9761	0.9935	1047.0
50	0.9865	0.9704	0.9697	0.9701	0.9936	1160.5

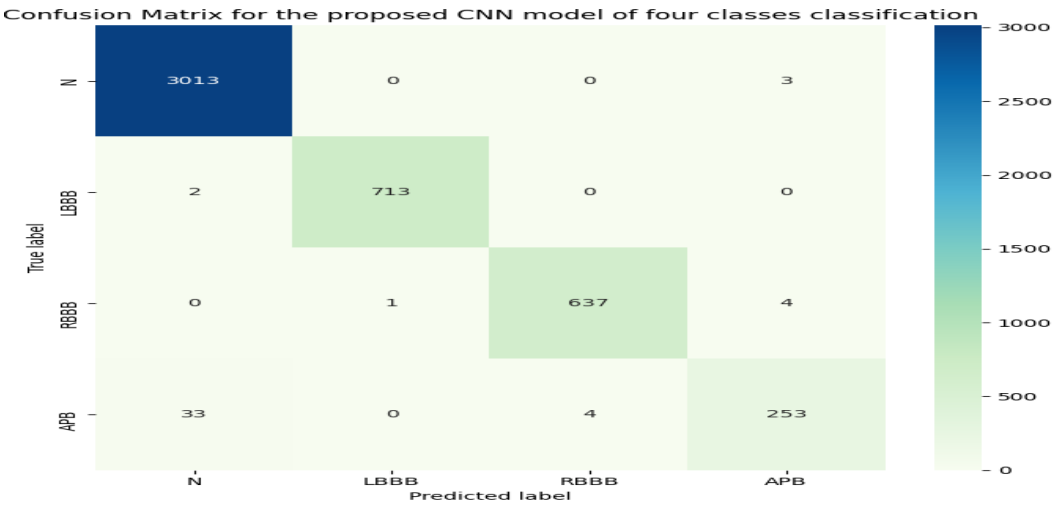


Fig. (4.12): Confusion matrix of the proposed CNN model for four-class classification with half normal datasets.

Other experiments have been implemented with four-class classification with 8000 and 6000 normal datasets and the results obtained from these experiments, in addition to the training time for each epoch in seconds, are presented in table (4-19) and (4-20) for 8000 and 6000 normal dataset experiments, respectively.

Table (4-19): The performance of four-class classification with 8000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9804	0.9712	0.9535	0.9623	0.9924	81.7
10	0.9807	0.9766	0.9490	0.9626	0.9917	168.2
15	0.9865	0.9808	0.9689	0.9748	0.9944	250.4
20	0.9859	0.9804	0.9654	0.9728	0.9944	334.6
25	0.9859	0.9777	0.9679	0.9728	0.9946	420.7
30	0.9896	0.9827	0.9763	0.9795	0.9960	624.7
35	0.9853	0.9793	0.9651	0.9722	0.9940	711.5
40	0.9859	0.9758	0.9689	0.9723	0.9947	886.1
45	0.9911	0.9833	0.9825	0.9829	0.9967	828.9
50	0.9893	0.9848	0.9730	0.9788	0.9957	831.4

Table (4-20): The performance of four-class classification with 6000 normal dataset						
Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9776	0.9578	0.9691	0.9634	0.9920	70.2
10	0.9808	0.9706	0.9651	0.9679	0.9933	145.8
15	0.9818	0.9807	0.9593	0.9699	0.9930	219.9
20	0.9857	0.9852	0.9680	0.9766	0.9945	290.6
25	0.9745	0.9570	0.9583	0.9576	0.9914	356.7
30	0.9871	0.9836	0.9740	0.9788	0.9953	615.5
35	0.9749	0.9732	0.9550	0.9640	0.9907	726.9
40	0.9871	0.9838	0.9743	0.9790	0.9952	829.6
45	0.9874	0.9861	0.9741	0.9800	0.9951	923.9
50	0.9867	0.9834	0.9729	0.9781	0.9950	939.1

the confusion matrices of the proposed CNN model for the four-class classification with 8000 and 6000 normal datasets of the experiment at epoch 45 have presented in fig. (4.13) and fig.(4.14), respectively.

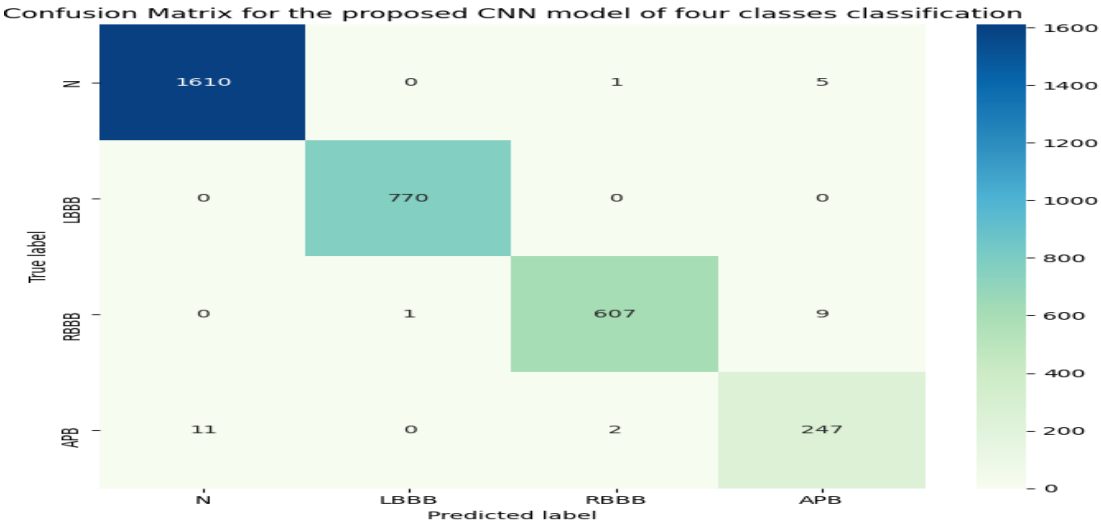


Fig. (4.13): Confusion matrix of the proposed CNN model for four-class classification with 8000 normal datasets.

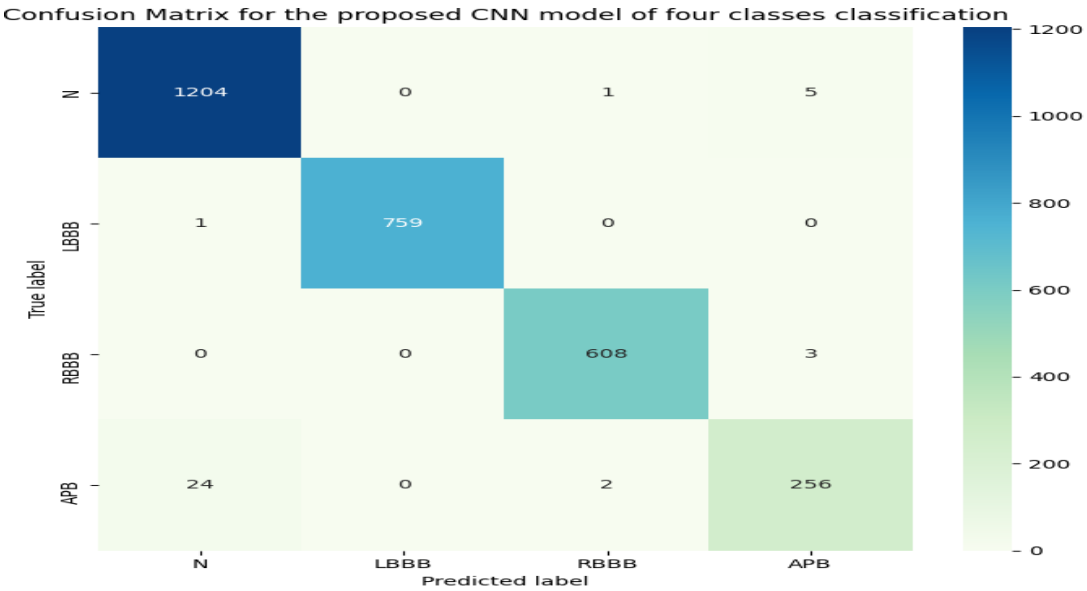
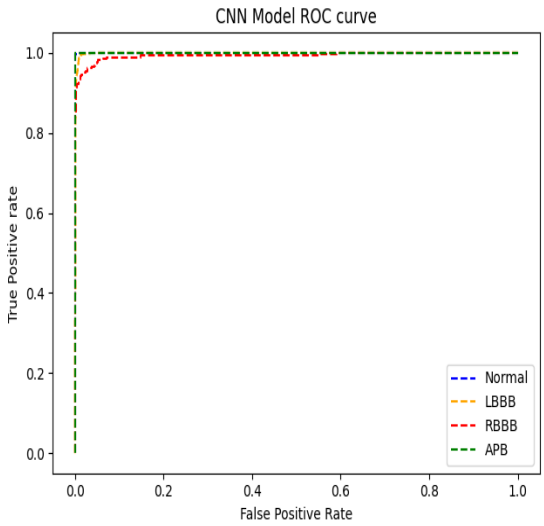
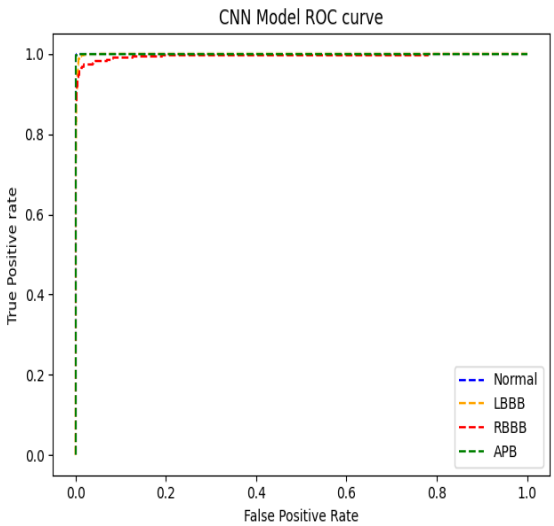


Fig. (4.14): Confusion matrix of the proposed CNN model for four-class classification with 6000 normal datasets.

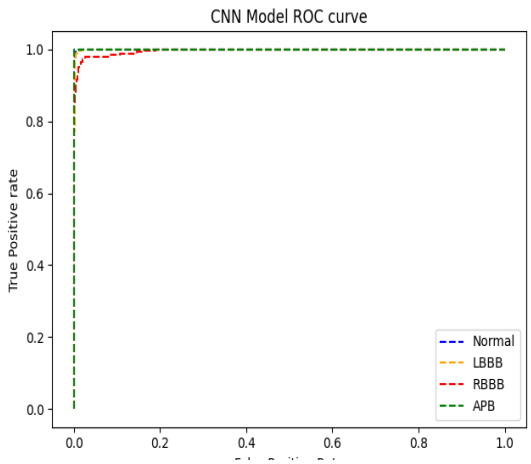
The ROC-curves of the proposed model for the four-class classification with full, half, 8000, and 6000 normal datasets have been represented in fig. (4.15) (a) (b) (c) (d), respectively.



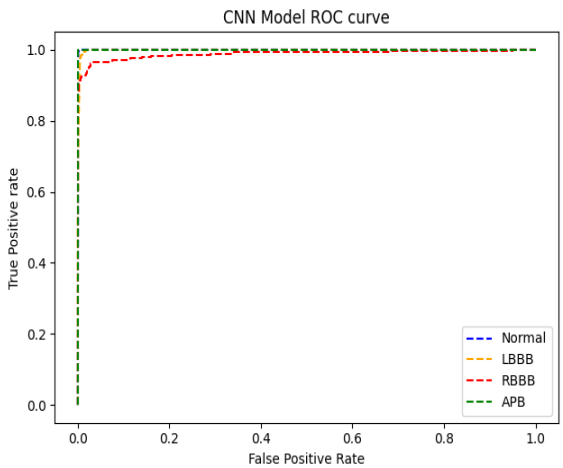
(a)



(b)



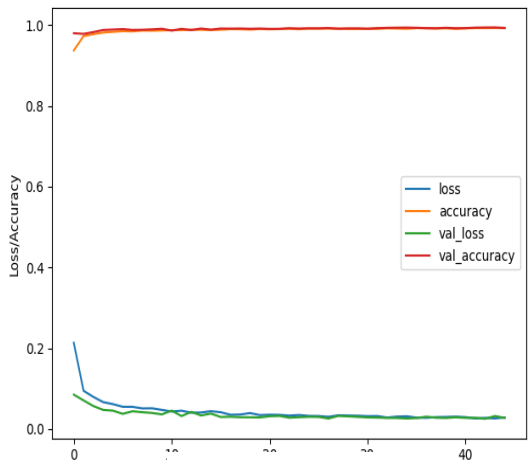
(c)



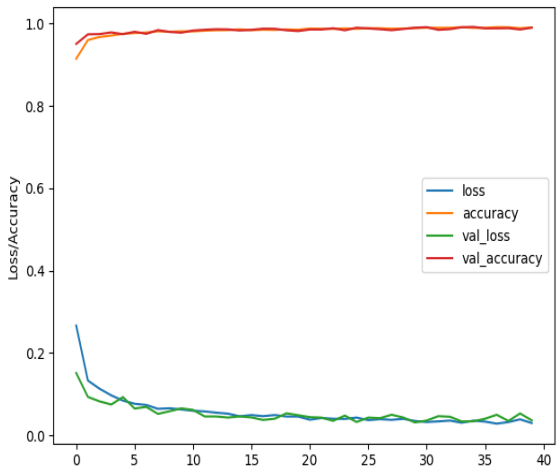
(d)

Fig. (4.15): ROC-curves of proposed model for four-class classification with (a)full normal datasets, (b) half normal dataset, (c) 8000 normal dataset (d) 6000 normal dataset

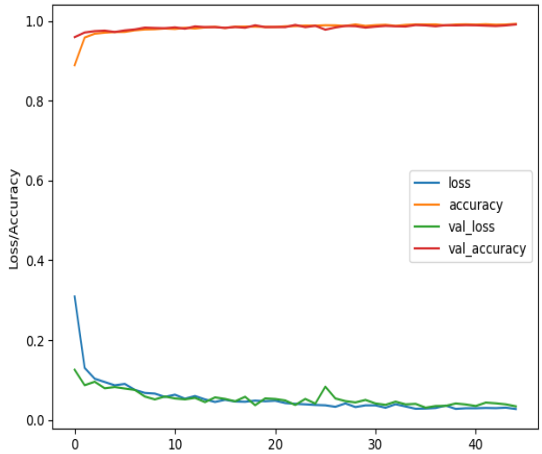
The loss, accuracy, validation-loss, and validation-accuracy of the proposed model for the four-class classification with full, half, 8000, and 6000 normal datasets have been represented in fig. (4.16) (e) (f) (g) (h), respectively.



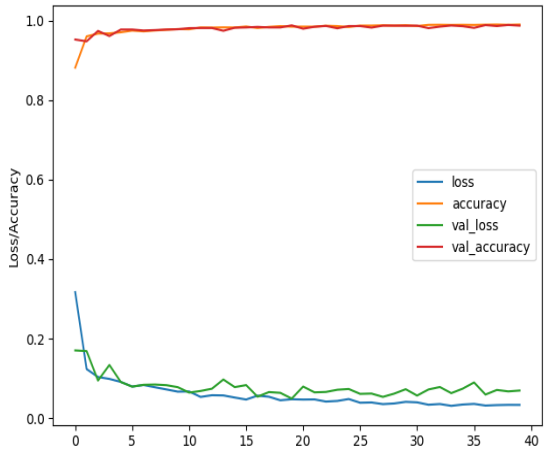
(e)



(f)



(g)



(h)

Fig. (4.16): Loss, accuracy, validation loss, and validation accuracy of proposed model for four-class classification with (e)full normal datasets, (f) half normal dataset, (g) 8000 normal dataset (h) 6000 normal dataset.

4.5.4 Six-class classification

In six-class classification, six different classes of arrhythmia are diagnosed by the proposed algorithm, named, N, LBBB, RBBB, APB, PVC, and F. The amount of normal dataset has been reduced as previous experiments with binary and four-class classification and for each amount of normal dataset, ten experiments with varied epoch numbers at each experiment have been implemented. The performance of six-class classification with overall normal dataset has been shown in table (4-21).

Table (4-21): The performance of six-class classification with full normal dataset						
Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9843	0.9642	0.9086	0.9356	0.9922	281.3
10	0.9813	0.9401	0.9128	0.9262	0.9908	582.7
15	0.9879	0.9740	0.9242	0.9484	0.9933	834.5
20	0.9882	0.9518	0.9471	0.9495	0.9947	1167.1
25	0.9901	0.9721	0.9513	0.9616	0.9949	1336.2
30	0.9893	0.9691	0.9419	0.9553	0.9947	1734.2
35	0.9886	0.9679	0.9281	0.9476	0.9941	1364.0
40	0.9900	0.9617	0.9551	0.9584	0.9955	2309.4
45	0.9887	0.9636	0.9350	0.9491	0.9946	1844.6
50	0.9896	0.9732	0.9387	0.9556	0.9951	2091.5

The confusion matrix of the proposed CNN model for the six-class classification with full normal datasets of the experiment at epoch 40 has shown in fig. (4.17).

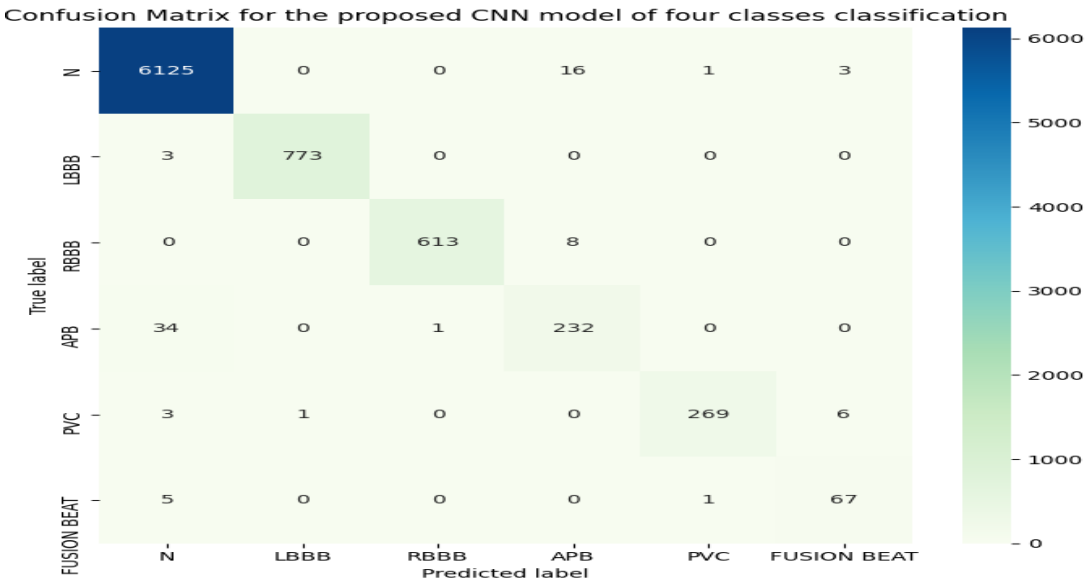


Fig. (4.17): Confusion matrix of the proposed CNN model for six-class classification with full normal dataset.

In six-class classification with half normal dataset, the performance of the ten experiments have been shown in table (4-22).

Table (4-22): The performance of six-class classification with 15000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9700	0.9582	0.8706	0.9123	0.9901	129.6
10	0.9768	0.9472	0.9104	0.9284	0.9931	268.3
15	0.9814	0.9720	0.9331	0.9522	0.9936	359.7
20	0.9840	0.9719	0.9380	0.9546	0.9947	492.2
25	0.9786	0.9507	0.9249	0.9376	0.9929	613.5
30	0.9852	0.9787	0.9355	0.9566	0.9951	853.8
35	0.9854	0.9667	0.9530	0.9598	0.9952	826.6
40	0.9954	0.9725	0.9436	0.9578	0.9949	1140.1
45	0.9854	0.9576	0.9607	0.9591	0.9956	1135.3
50	0.9850	0.9675	0.9516	0.9595	0.9951	1623.8

The confusion matrix of the proposed CNN model for the six-class classification with half normal datasets with experiment at epoch 45 has been shown in fig. (4.18).

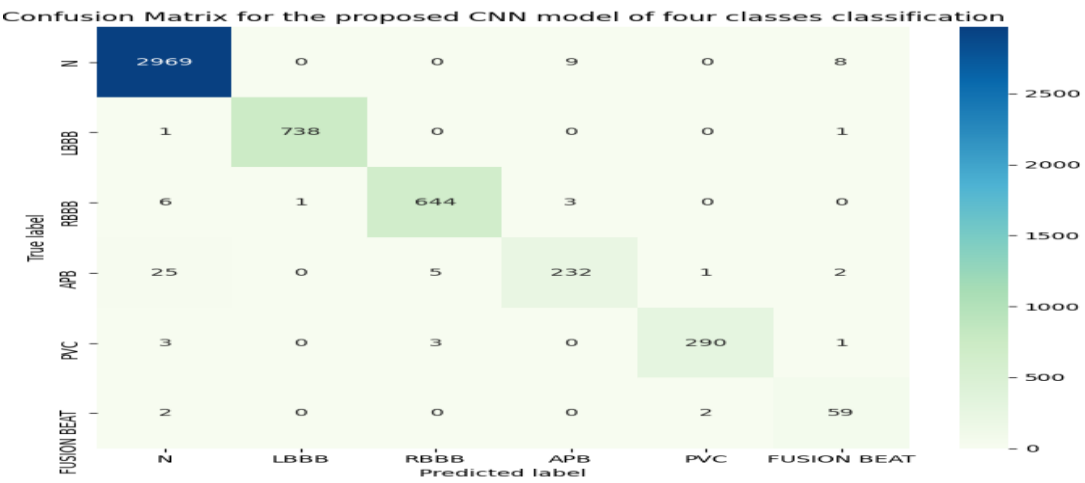


Fig. (4.18): Confusion matrix of proposed CNN model for six-class classification with half normal datasets.

The performances of six-class classification with 8000 normal datasets and the best results which are obtained from last experiment with epoch 50 are presented in table (4-23) and the confusion matrix is illustrated in fig. (4.19).

Table (4-23): The performance of six-class classification with 8000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9798	0.9604	0.9591	0.9597	0.9952	92.4
10	0.9864	0.9735	0.9722	0.9728	0.9970	177.4
15	0.9867	0.9797	0.9600	0.9697	0.9967	828.6
20	0.9867	0.9676	0.9713	0.9694	0.9969	1084.3
25	0.9867	0.9770	0.9628	0.9699	0.9968	1494.1
30	0.9900	0.9812	0.9756	0.9784	0.9977	1164.8
35	0.9900	0.9776	0.9765	0.9771	0.9978	638.8
40	0.9870	0.9756	0.9740	0.9748	0.9971	707.8
45	0.9859	0.9682	0.9784	0.9733	0.9971	769.4
50	0.9914	0.9858	0.9831	0.9844	0.9980	1014.6

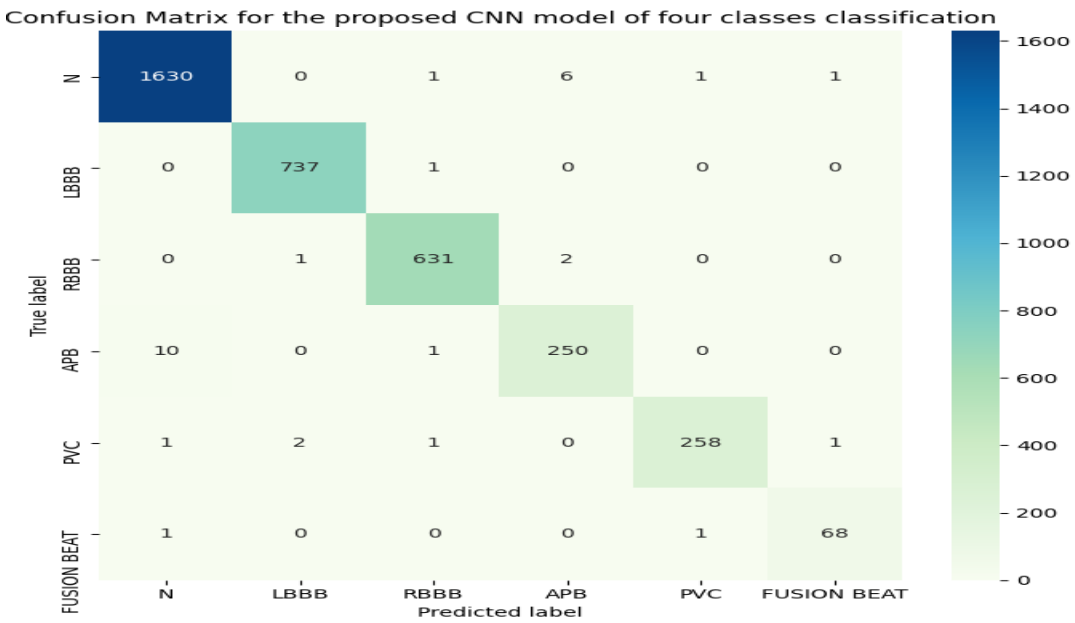


Fig. (4.19): Confusion matrix of the CNN model for six-class classification with 8000 normal datasets.

Lastly, in six-class classification with 6000 normal, the performances of all experiments and the confusion matrix of the proposed model for experiment with highest results at epoch 50 are shown in table (4-24) and fig. (4.20), respectively.

Table (4-24): The performance of six-class classification with 6000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training Time in sec.
5	0.9791	0.9718	0.9585	0.9651	0.9952	80.868
10	0.9813	0.9747	0.9688	0.9718	0.9958	159.088
15	0.9697	0.9555	0.9545	0.9550	0.9935	243.077
20	0.9838	0.9792	0.9526	0.9657	0.9964	317.748
25	0.9863	0.9777	0.9758	0.9768	0.9971	403.095
30	0.9866	0.9762	0.9730	0.9746	0.9970	469.496
35	0.9866	0.9775	0.9760	0.9767	0.9971	744.030
40	0.9847	0.9757	0.9707	0.9732	0.9967	635.110
45	0.9863	0.9789	0.9762	0.9775	0.9972	719.879
50	0.9900	0.9863	0.9783	0.9823	0.9977	814.1

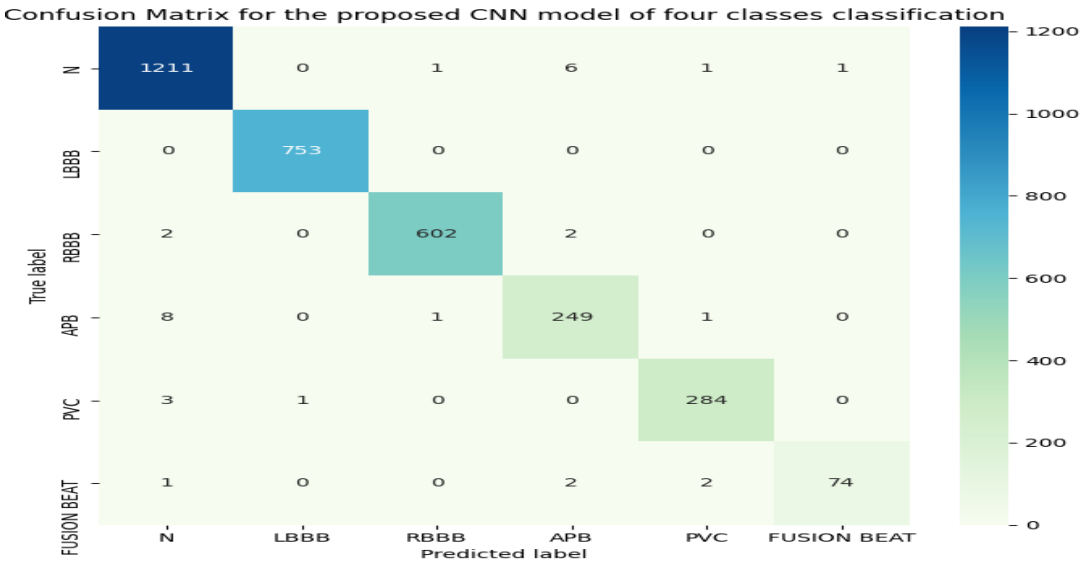


Fig. (4.20): Confusion matrix of proposed CNN model for six-class classification with 6000 normal datasets.

The ROC-curves of the proposed model for the six-class classification with full, half, 8000, and 6000 normal datasets have been represented in fig. (4.21) (a) (b) (c) (d), respectively.

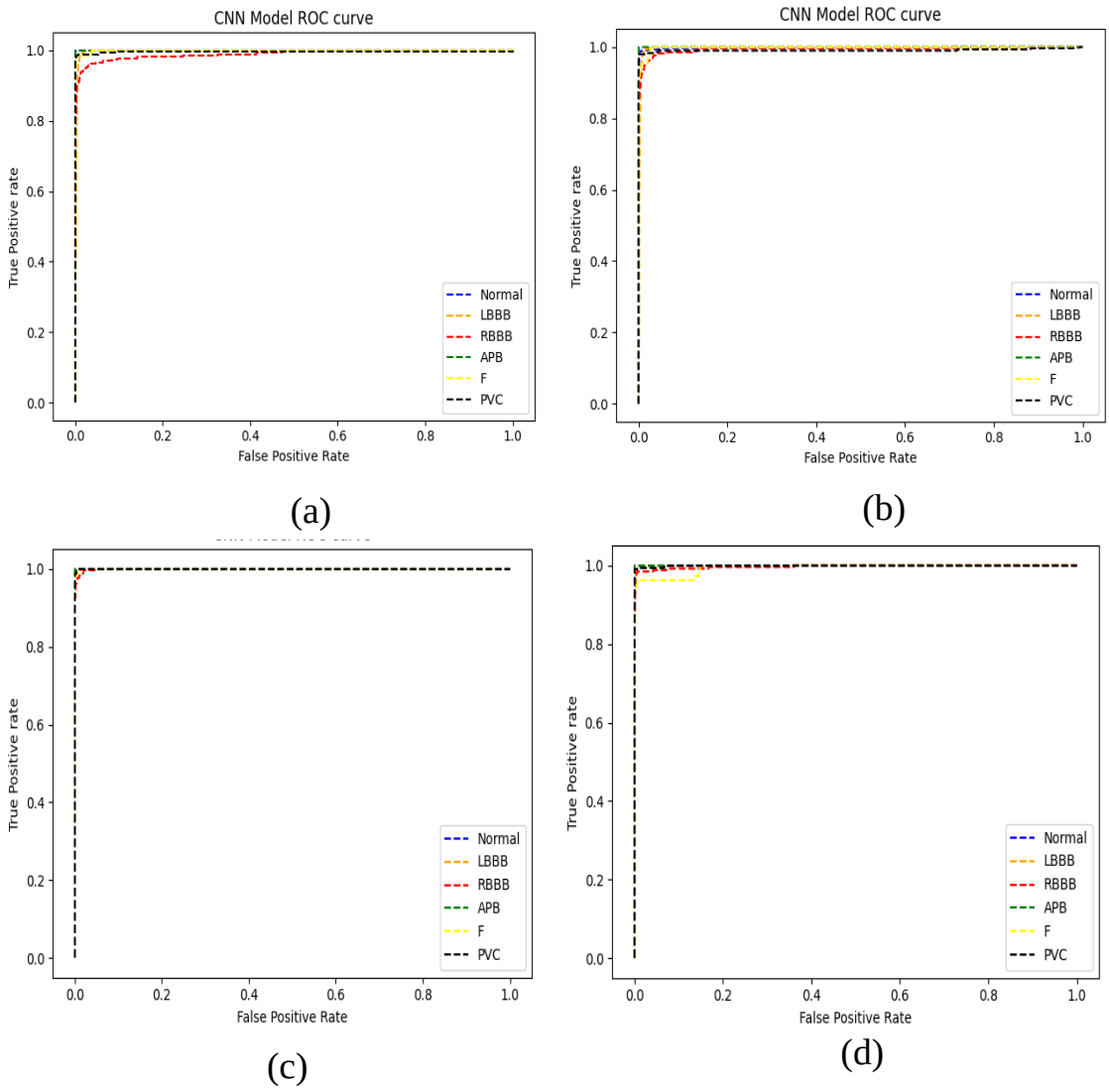


Fig. (4.21): ROC-curves of proposed model for six-class classification with (a)full normal Datasets, (b) half normal dataset, (c) 8000 normal dataset (d) 6000 normal dataset

The loss, accuracy, validation-loss, and validation-accuracy of the proposed model for the six-class classification with full, half, 8000, and 6000 normal datasets are represented in Fig. (4.22) (e) (f) (g) (h), respectively.

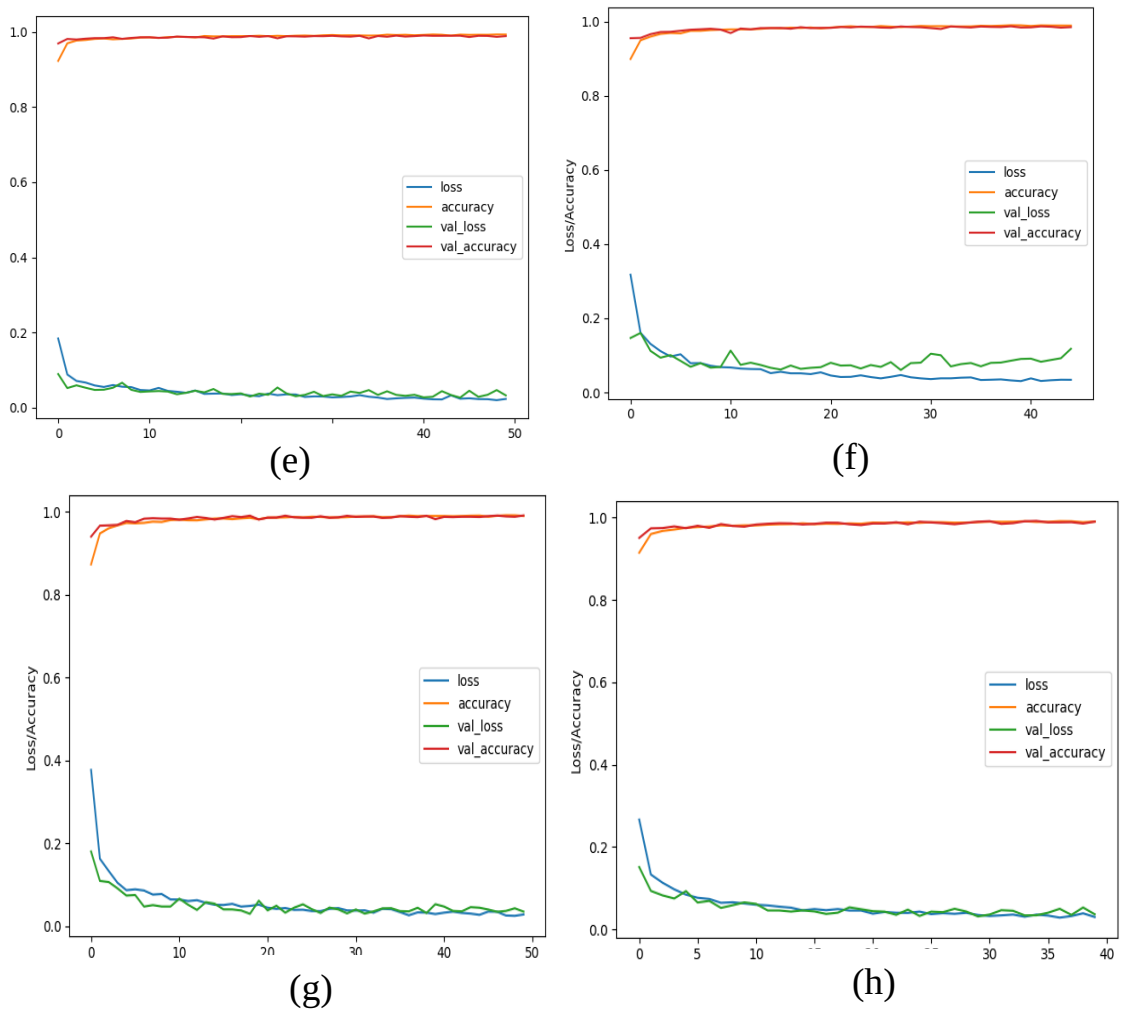


Fig. (4.22): Loss, accuracy, validation loss, and validation accuracy of proposed model for six-class classification with (e)full normal datasets, (f) half normal dataset, (g) 8000 normal dataset (h) 6000 normal dataset.

4.5.5 Seven-class classification

In seven-class classification, seven different classes of arrhythmia are diagnosed by the proposed algorithm, named, N, LBBB, RBBB, APB, PVC, F, and P. The amount of normal dataset has been reduced as previous experiments with for each amount of normal dataset. Ten experiments with varied epoch numbers have been implemented. Firstly, the experiments of seven-class classification have been implemented with overall normal

dataset and the performances of those experiments are shown in table (4-25). Also, the confusion matrix of the experiment at epoch 50 is shown in fig. (4.23).

Table (4-25): The performance of seven-class classification with full normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training Time in sec.
5	0.9778	0.9488	0.9012	0.9244	0.9928	253.175
10	0.9864	0.9682	0.9459	0.9569	0.9953	443.047
15	0.9857	0.9739	0.9263	0.9495	0.9945	652.912
20	0.9873	0.9683	0.9507	0.9594	0.9957	884.733
25	0.9859	0.9534	0.9517	0.9526	0.9958	1114.699
30	0.9872	0.9583	0.9580	0.9582	0.9959	2764.856
35	0.9894	0.9735	0.9559	0.9646	0.9965	4809.576
40	0.9882	0.9665	0.9586	0.9625	0.9963	1760.236
45	0.9893	0.9786	0.9429	0.9605	0.9960	2003.933
50	0.9908	0.9728	0.9646	0.9687	0.9969	3909.1

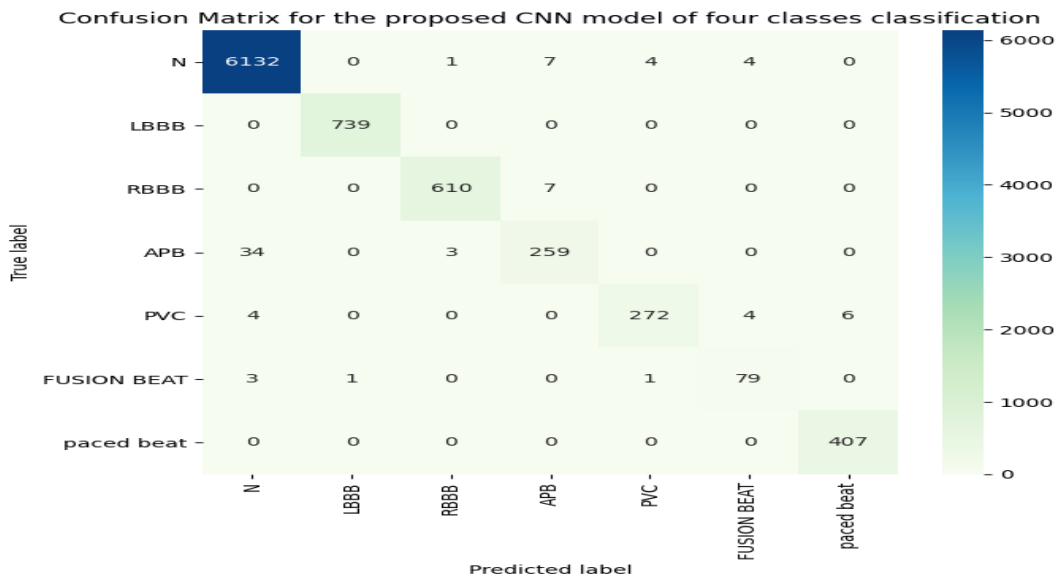


Fig. (4.23): Confusion matrix of proposed CNN model for seven-class classification with full normal dataset.

In seven-classes classification with half normal dataset (15000 sequences), the experiments' performances and the best experiment results have been presented in table (4-26). The confusion matrix of experiment at epoch 50 is illustrated in fig. (4.24).

Table (4-26): The performance of seven-class classification with 15000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec%	Training time in sec.
5	0.9744	0.9715	0.9184	0.9442	0.9933	183.1
10	0.9825	0.9586	0.9575	0.9580	0.9961	295.4
15	0.9786	0.9656	0.9346	0.9499	0.9949	601.8
20	0.9819	0.9651	0.9456	0.9553	0.9957	526.5
25	0.9832	0.9729	0.9432	0.9578	0.9961	760.6
30	0.9823	0.9702	0.9502	0.9601	0.9955	1138.4
35	0.9845	0.9697	0.9616	0.9656	0.9964	948.2
40	0.9838	0.9602	0.9507	0.9515	0.9954	1228.2
45	0.9801	0.9523	0.9631	0.9690	0.9973	2992.3
50	0.9880	0.9680	0.9666	0.9673	0.9974	1532.3

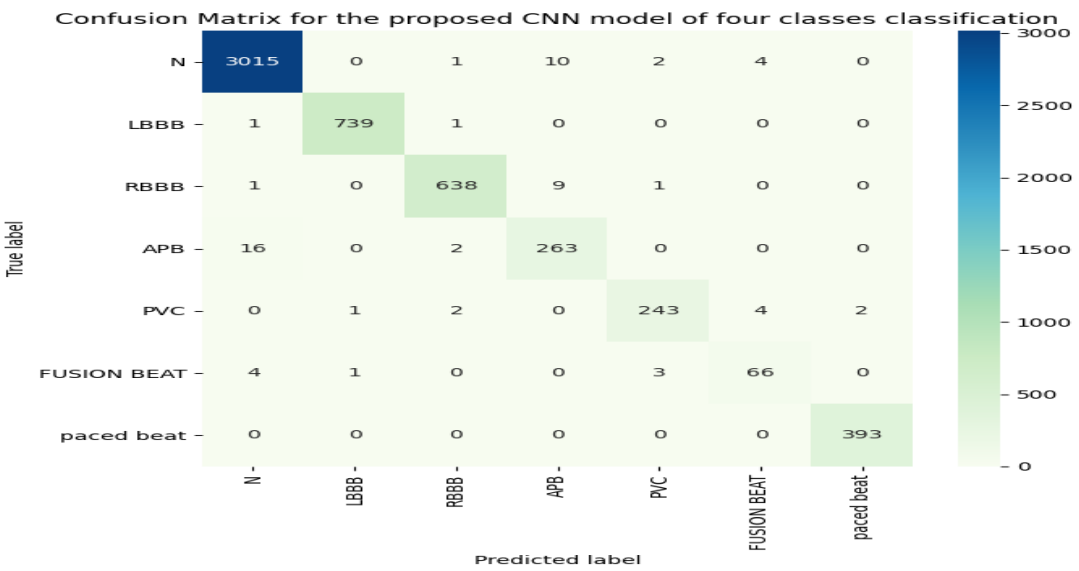


Fig. (4.24): Confusion matrix of the proposed CNN model for seven-class classification with half normal datasets.

Finally, the performances of the seven-classes experiment with 8000 and 6000 normal datasets are shown in table (4-27) and table (4-28), respectively. Also, the confusion matrices for the seven-class classification with 8000 and 6000 normal datasets are show in fig. (4.25) and fig. (4.26), respectively.

Table (4-27): The performance of seven-class classification with 8000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1%	Spec%	Training time in sec.
5	0.9704	0.9651	0.9256	0.9449	0.9942	98.2
10	0.9766	0.9599	0.9529	0.9564	0.9955	308.1
15	0.9799	0.9748	0.9592	0.9670	0.9961	421.8
20	0.9828	0.9718	0.9710	0.9714	0.9967	576.3
25	0.9801	0.9681	0.9621	0.9651	0.9964	752.6
30	0.9814	0.9588	0.9670	0.9629	0.9967	900.9
35	0.9848	0.9753	0.9664	0.9708	0.9972	782.7
40	0.9816	0.9646	0.9646	0.9646	0.9967	1204.2
45	0.9841	0.9690	0.9717	0.9704	0.9972	1914.1
50	0.9858	0.9740	0.9723	0.9731	0.9973	1035.3

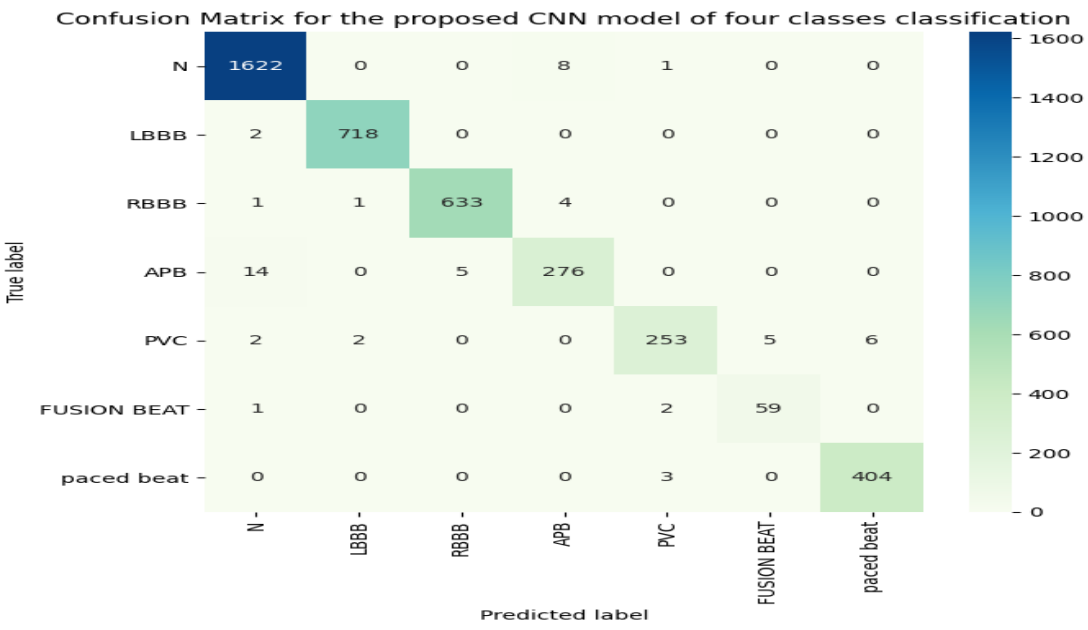


Fig. (4.25): Confusion matrix of the proposed CNN model for seven-class classification with 8000 normal datasets.

Table (4-28): The performance of seven-class classification with 6000 normal dataset

Metric / Epoch no.	Acc %	Pre %	Sen %	F1 %	Spec %	Training time in sec.
5	0.9732	0.9463	0.9497	0.9480	0.9950	115.5
10	0.9735	0.9590	0.9470	0.9529	0.9953	239.2
15	0.9798	0.9734	0.9640	0.9687	0.9963	302.9
20	0.9829	0.9717	0.9718	0.9717	0.9970	352.2
25	0.9779	0.9651	0.9666	0.9658	0.9960	664.3
30	0.9771	0.9625	0.9628	0.9627	0.9958	554.3
35	0.9793	0.9657	0.9693	0.9675	0.9964	739.3
40	0.9859	0.9832	0.9734	0.9783	0.9973	747.3
45	0.9823	0.9717	0.9695	0.9706	0.9968	857.8
50	0.9821	0.9616	0.9700	0.9658	0.9968	877.3

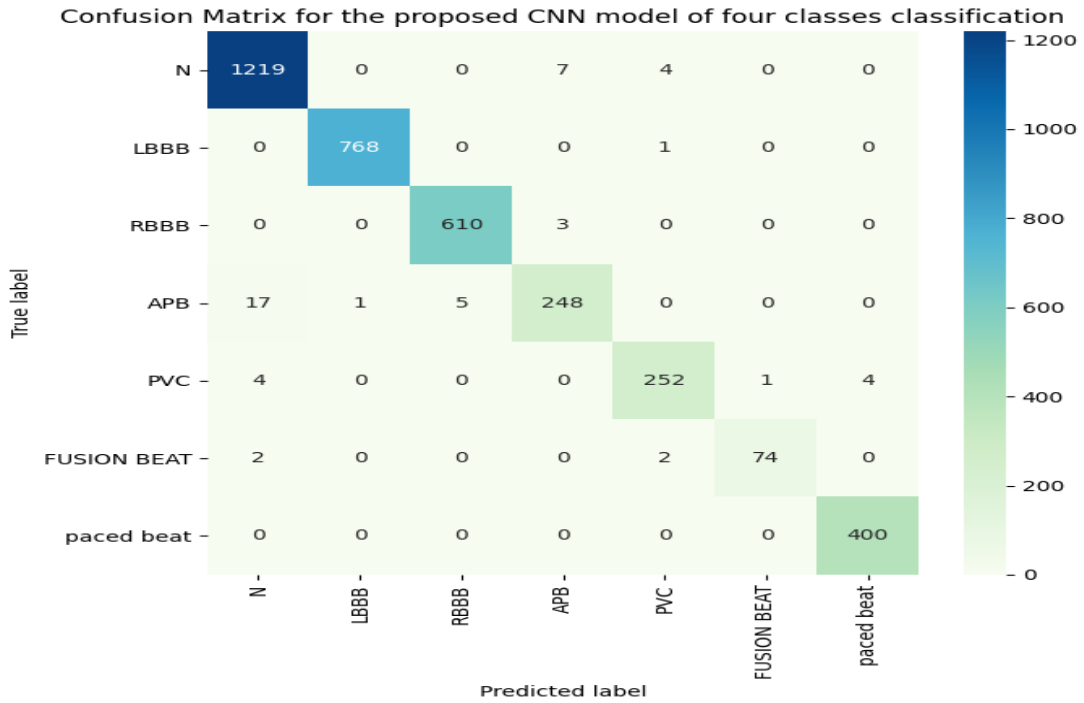
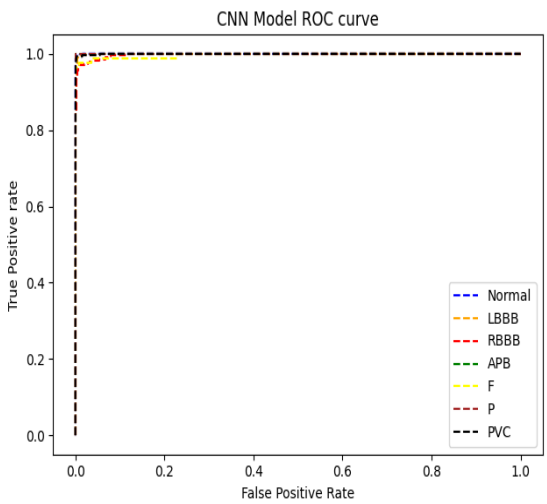
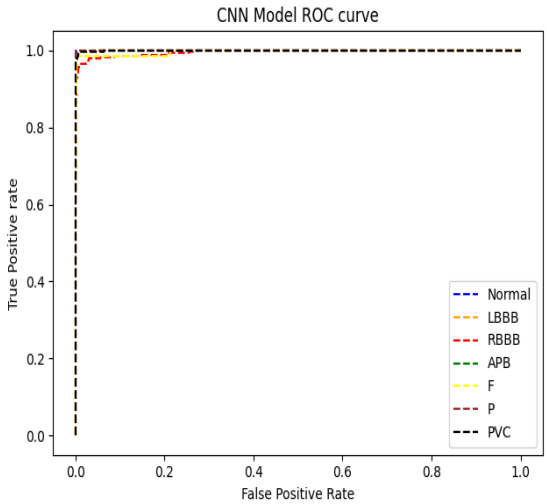


Fig. (4.26): confusion matrix of the proposed CNN model for seven-class classification with 6000 normal datasets.

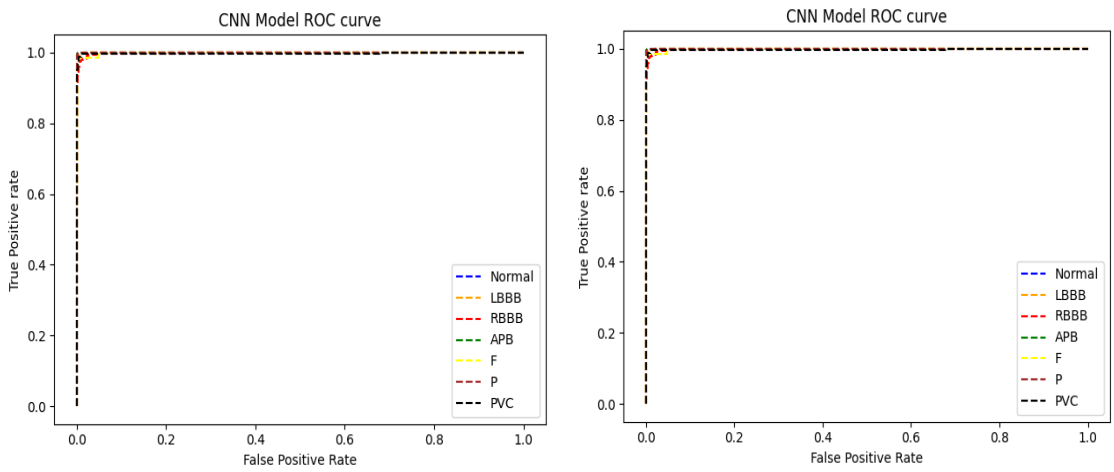
The ROC-curves of the proposed model for the seven-class classification with full, half, 8000, and 6000 normal datasets are represented in fig. (4.27) (a) (b) (c) (d), respectively.



(a)



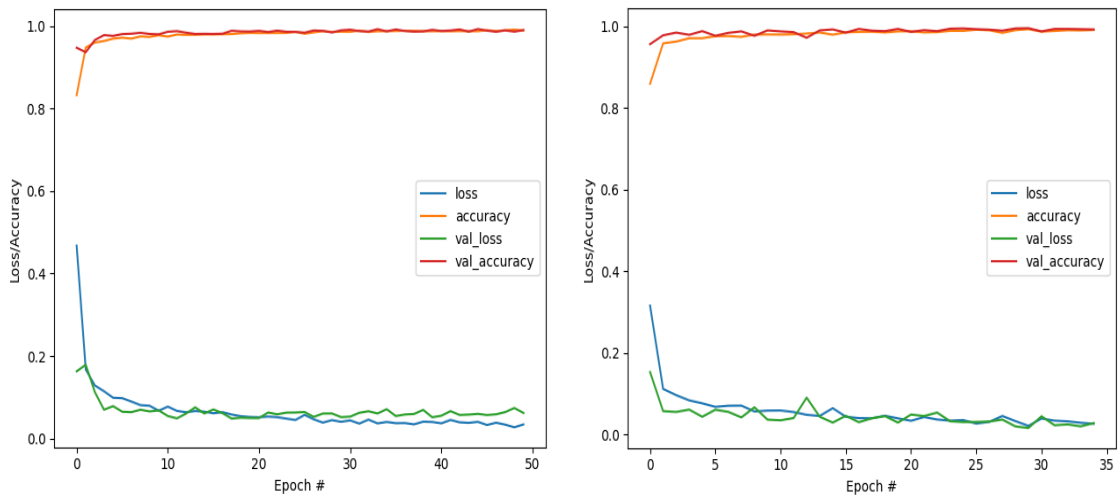
(b)



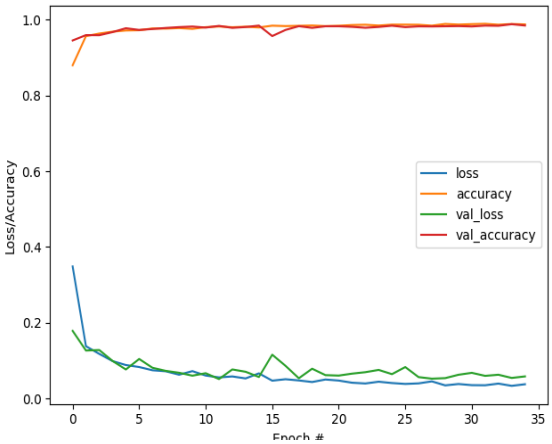
(c) (d)

Fig. (4.27): Roc-curves of proposed model for seven-class classification with (a)full normal datasets, (b) half normal dataset, (c) 8000 normal dataset (d) 6000 normal dataset

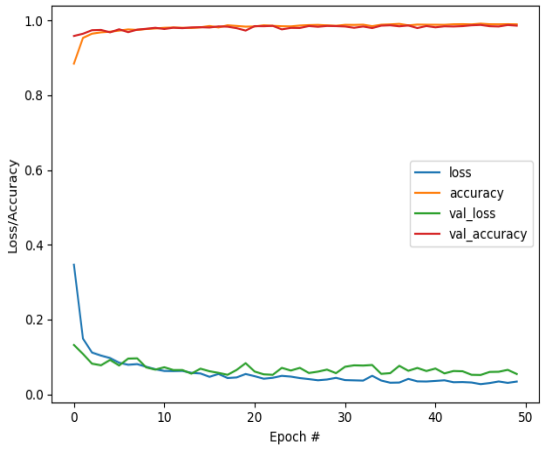
The loss, accuracy, validation-loss, and validation-accuracy of the proposed model for the seven-class classification with full, half, 8000, and 6000 normal datasets are represented in Fig. (4.28) (e) (f) (g) (h), respectively.



(e) (f)



(g)



(h)

Fig. (4.28): Loss, accuracy, validation loss, and validation accuracy of proposed model for seven-class classification with (e)full Normal Datasets, (f) half normal dataset, (g) 8000 normal dataset (h) 6000 normal dataset.

4.5.6 Experimental results for different partitioning of the training and test dataset

For the proposed CNN model, a set of experiments were applied with another range of the training and testing dataset in order to examine the ability of the proposed model and to identify the most appropriate partitioning parameters. For each classification scheme, the dataset was divided into a 70% training set and 30% testing set with experimental results illustrated in table (4-29).

Table (4-29): Experimental results of the proposed model for all classification schemes with different training and testing dataset partitioning

Epoch	Classification scheme	Acc%	Pre%	Sen%	Spec%	F1%	Training time in sec
45	Binary	98.86	98.77	98.48	98.48	98.62	2025.4
35	Three-class	99.32	99.01	98.96	99.67	98.98	274.6
45	Four-class	98.88	98.62	97.19	99.54	97.90	640.2
50	Six-class	98.91	98.23	97.30	99.74	97.76	825.2
40	Seven-class	98.55	97.87	97.27	99.74	97.57	659.3

4.6 Discussion

In this thesis, to demonstrate the effectiveness of the proposed 1-D CNN model, many experiments were performed on the arrhythmia data set and were applied using different parameters randomly until reached the appropriate parameters for the proposed model that can achieve the best results. such as

- Sequence length of 128 samples,
- Batch size of 20,
- Epoch number of 30 - 50,
- Kernel size of 5
- Stride of (1,2)
- Best dataset splitting range of 80% training set and 20% testing set for all classification types.

The proposed CNN model, using 1-D signals, can classify different types of arrhythmia using various classification schemes which are binary and multi-class classification. In binary classification the proposed model has detected the normal from abnormal class. While, in multi-class classification, seven various types of ECG heartbeats namely, N, LBBB, RBBB, APB, PVC, P, and F are classified by utilizing different classification methods such as three-, four-, six, and seven-class classification. The selection of arrhythmia's classes was based on their influence and popularity, as well as some of them considered life-threatening.

To give a more accurate presentation about the proposed model, the performance of the proposed CNN model was compared with previous

arrhythmia classification studies. Because the number of training and test sets, CNN model structures, and number of arrhythmia categories used for detection and classification varied in these studies, other metrics besides accuracy are used to make accurate comparisons. However, when compared to other prior researches, the suggested CNN model outperformed them while offering an efficient technique to identifying ECG arrhythmia using one-dimensional data. It should be notable that different techniques have been used 1-D data for arrhythmia classification.

Table (4-30) below summarizes a comparison of the proposed model to the state-of-the-art approaches in terms of the number of classes, evaluation metrics, the model employed, and the database. Among these techniques, the researchers in [16] have presented a classification system of CVDs using CNN and MLP network. This study showed lower performance than the proposed model in both using the CNN network and the MLP network, i.e., 83.5 and 88.7, respectively. While the proposed model has achieved the highest results, especially in classifying two classes with an accuracy of 99.27%, In multi-category classification, the proposed 1-D CNN model outperformed models [11], [13], and [15] in terms of the performance achieved in accuracy, sensitivity, and specificity, as well as the number of categories (in a six-category classification), where the proposed model classified six different categories, while the mentioned models were able to classify only five types of arrhythmias with lower performance than the proposed model. In [12], [7], and [19] models were presented to classify four

classes with performances less than the proposed model for the same number of classes.

Table (4-30): Comparison of the proposed CNN model with recent arrhythmia classification techniques.

Researcher and year	Classes	Acc %	Sen %	Spec %	Pre %	F1%	Model	Database
Zubair et al., 2016 [11]	5	92.7					4-layer 1-D CNN	MITDB
Acharya, Fujita et al., 2017[12]	4	94.90	99.13	81.44			11-layer 1-D CNN	MITDB AFDB CUDB
Acharya et al., 2017 [13]	5	94.03	96.71	96.71			9- layer CNN	MITDB
Oh et al., 2018 [15]	5	98.10	97.50	98.70			CNN+LSTM	MITDB
Savalia & Emamian, 2018 [16]	9(CNN) 2(MLP)	83.5 CNN 88.7 MLP					5-layer 1-D CNN	MITDB
W. Zhang et al., (2018)[7]	4		97.6		97.7	97.6	12-layer 1D-CNN	MITDB
Petmezas et al., 2021[19]	4		97.87	99.29			8-layer 1D CNN–LSTM	AFDB
Proposed CNN model	2 3 4 6 7	99.27 99.52 99.11 99.14 98.59	98.98 99.26 98.25 98.31 97.34	98.98 99.76 99.67 99.80 99.73	99.26 99.34 98.33 98.58 98.32	99.12 99.30 98.29 98.44 97.83	9-layer 1-D CNN	MITDB

From Table (4-30), the results achieved for the proposed model when compared with the recent studies to classify arrhythmias automatically showed that the proposed 1-D CNN algorithm had the highest results in terms of accuracy, accuracy, sensitivity, specificity, and F1-score compared to the state-of-the-art models.

Also, the proposed model has yielded the highest performance in the three-class classification scheme, which is inevitably due to the balance of the arrhythmia dataset that were classified. It can be considered the preferred classification type because the proposed model was able to achieve optimal results with a very small amount of data and a relatively low training time. It should be noted that the imbalance of data in some experiments did not affect the efficiency of the model, and this indicates the robust and stability of the model which can achieve the highest results in both balance and imbalance datasets.

It's worth mentioning that identifying such cardiac arrhythmias by a clinical physician is a time-consuming task that might last for hours. By recognizing these patterns and prompting the observers to look more attentively at locations of greater relevance, the artificial neural network-based system might improve the performance of clinical physicians, which might help with the diagnosis and therapy of some of the most serious cardiovascular diseases.

Chapter 5 – Conclusions and Future Works

5.1 Conclusions

In this thesis, the most important characteristics that have been discovered during the design and implementation of the proposed system and achieving it is results are as follows:

1. The presented a 1-D CNN-based classification algorithm for automated detection and classification of heart rhythm disorders employing 1-D ECG signals. In the diagnosis and prevention of CVDs, a precise taxonomy for Electrocardiogram signals is extremely useful.
2. The proposed approach showed a high level of accuracy despite using a simple nine-layer hierarchy, small epochs and little training time.
3. This model can detect cardiac diseases through binary-classification of (normal and abnormal classes) with a high performance. At the same time, it can classify seven classes of arrhythmia: N, LBBB, RBBB, APB, PVC, P, and F, with in the different classification schemes that are three-, four-, six-, and seven-class classification with the same high degree of performance, so it can be used in various fields such as monitoring abnormalities in ECG signals and supporting cardiologists in presenting an accurate decision about the arrhythmia class.
4. In this study, only MITDB dataset was used to test the suggested model. We extracted as much as possible the classes with the largest number of data set and the most influential for the purpose of using them in classification. A dataset with additional classes and samples would be able to better assess the proposed model on various types of heart disorders.

5.2 Future works

Some future works suggestions can be viewed below:

1. The produced research demonstrates promising results, indicating that it can make effective generalizations of the ECG signal, which is very useful for future employment of this framework to identify ECG irregularities by connecting the remote ECG monitoring system to more clients in the clinical routine.
2. Additional layers may be added to the hierarchy of the proposed model, which may enhance its effectiveness and performance.
3. The arrhythmia detection and classification method might also be enhanced, mostly by collecting more data from various forms of arrhythmia, allowing for the addition of more classes to identify a larger range of abnormal conditions. In other words, we might see more arrhythmia types.
4. More arrhythmia data preprocessing might be investigated in order to maintain the key features of the arrhythmias, such as noise removing techniques.
5. More arrhythmia databases might be used to test the classifier and compare various experimental results to correctly evaluate the performance of the model.
6. Suggesting other heart diseases to be classified by the proposed model.

References

[1] S. S and A. V, "Early Detection Of Life-Threatening Cardiac Arrhythmias Using Deep Learning Techniques," *Curr. Signal Transduct. Ther.*, vol. 14, pp. 1–12, 2019, doi: 10.2174/1574362414666191022145259.

[2] T. J. Jun, H. M. Nguyen, D. Kang, D. Kim, D. Kim, and Y. H. Kim, "ECG arrhythmia classification using a 2-D convolutional neural network," *arXiv*, 2018.

[3] B. M. Mathunjwa, Y. T. Lin, C. H. Lin, M. F. Abbod, and J. S. Shieh, "ECG arrhythmia classification by using a recurrence plot and convolutional neural network," *Biomed. Signal Process. Control*, vol. 64, no. February 2020, p. 102262, 2021, doi: 10.1016/j.bspc.2020.102262.

[4] A. Ullah, S. M. Anwar, M. Bilal, and R. M. Mehmood, "Classification of arrhythmia by using deep learning with 2-D ECG spectral image representation," *Remote Sens.*, vol. 12, no. 10, pp. 1–14, 2020, doi: 10.3390/rs12101685.

[5] J. Huang, B. Chen, B. Yao, and W. He, "ECG Arrhythmia Classification Using STFT-Based Spectrogram and Convolutional Neural Network," *IEEE Access*, vol. 7, pp. 92871–92880, 2019, doi: 10.1109/ACCESS.2019.2928017.

[6] R. He *et al.*, "Automatic Cardiac Arrhythmia Classification Using Combination of Deep Residual Network and Bidirectional LSTM," *IEEE Access*, vol. 7, pp. 102119–102135, 2019, doi: 10.1109/ACCESS.2019.2931500.

[7] W. Zhang, L. Yu, L. Ye, W. Zhuang, and F. Ma, "ECG Signal Classification with Deep Learning for Heart Disease Identification," *Int. Conf. Big Data Artif. Intell. BDAI 2018*, no. June, pp. 47–51, 2018, doi:

10.1109/BDAI.2018.8546681.

[8] N. Zhang, S. L. Shen, A. Zhou, and Y. S. Xu, “Investigation on Performance of Neural Networks Using Quadratic Relative Error Cost Function,” *IEEE Access*, vol. 7, pp. 106642–106652, 2019, doi: 10.1109/ACCESS.2019.2930520.

[9] R. R. Van De Leur *et al.*, “Discovering and Visualizing Disease-Specific Electrocardiogram Features Using Deep Learning: Proof-of-Concept in Phospholamban Gene Mutation Carriers,” *Circ. Arrhythmia Electrophysiol.*, no. February, pp. 138–147, 2021, doi: 10.1161/CIRCEP.120.009056.

[10] G. B. Moody and R. G. Mark, “The impact of the MIT-BIH arrhythmia database,” *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, 2001, doi: 10.1109/51.932724.

[11] M. Zubair, J. Kim, and C. Yoon, “An automated ECG beat classification system using convolutional neural networks,” *2016 6th Int. Conf. IT Converg. Secur. ICITCS 2016*, 2016, doi: 10.1109/ICITCS.2016.7740310.

[12] U. R. Acharya, H. Fujita, O. S. Lih, Y. Hagiwara, J. H. Tan, and M. Adam, “Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network,” *Inf. Sci. (Ny).*, vol. 405, pp. 81–90, 2017, doi: 10.1016/j.ins.2017.04.012.

[13] U. R. Acharya *et al.*, “A deep convolutional neural network model to classify heartbeats,” *Comput. Biol. Med.*, vol. 89, pp. 389–396, 2017, doi: 10.1016/j.compbiomed.2017.08.022.

[14] A. Isin and S. Ozdalili, “Cardiac arrhythmia detection using deep learning,” *Procedia Comput. Sci.*, vol. 120, pp. 268–275, 2017, doi: 10.1016/j.procs.2017.11.238.

[15] S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya, “Automated

diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats," *Comput. Biol. Med.*, vol. 102, pp. 278–287, 2018, doi: 10.1016/j.compbiomed.2018.06.002.

[16] S. Savalia and V. Emamian, "Cardiac arrhythmia classification by multi-layer perceptron and convolution neural networks," *Bioengineering*, vol. 5, no. 2, 2018, doi: 10.3390/bioengineering5020035.

[17] Ö. Yildirim, "A novel wavelet sequences based on deep bidirectional LSTM network model for ECG signal classification," *Comput. Biol. Med.*, vol. 96, pp. 189–202, 2018, doi: 10.1016/j.compbiomed.2018.03.016.

[18] E. R. Onainor, "済無No Title No Title No Title," vol. 1, no. Dl, pp. 105–112, 2019.

[19] R. Avanzato and F. Beritelli, "Automatic ecg diagnosis using convolutional neural network," *Electron.*, vol. 9, no. 6, pp. 1–14, 2020, doi: 10.3390/electronics9060951.

[20] X. Wan, Z. Jin, H. Wu, J. Liu, B. Zhu, and H. Xie, "HEARTBEAT CLASSIFICATION ALGORITHM BASED on ONE-DIMENSIONAL CONVOLUTION NEURAL NETWORK," *J. Mech. Med. Biol.*, vol. 20, no. 7, pp. 1–15, 2020, doi: 10.1142/S0219519420500463.

[21] G. Petmezas *et al.*, "Automated Atrial Fibrillation Detection using a Hybrid CNN-LSTM Network on Imbalanced ECG Datasets," *Biomed. Signal Process. Control*, vol. 63, no. March 2020, p. 102194, 2021, doi: 10.1016/j.bspc.2020.102194.

[22] A. Ullah, S. U. Rehman, S. Tu, R. M. Mehmood, Fawad, and M. Ehatisham-Ul-haq, "A hybrid deep CNN model for abnormal arrhythmia detection based on cardiac ECG signal," *Sensors (Switzerland)*, vol. 21, no. 3, pp. 1–13, 2021, doi: 10.3390/s21030951.

[23] A. J. Weinhaus, "Anatomy of the human heart," *Handb. Card. Anatomy*,

Physiol. Devices, Third Ed., pp. 61–88, 2015, doi: 10.1007/978-3-319-19464-6_5.

[24] Y. Li and W. Cui, "Identifying the mislabeled training samples of ECG signals using machine learning," *Biomed. Signal Process. Control*, vol. 47, pp. 168–176, 2019, doi: 10.1016/j.bspc.2018.08.026.

[25] R. S. S. Kumari and J. G. Devi, "Classification of cardiac arrhythmias based on morphological and rhythmic features," *Int. J. Biomed. Eng. Technol.*, vol. 14, no. 3, pp. 192–208, 2014, doi: 10.1504/IJBET.2014.059669.

[26] T. A. Gaziano, "Cardiovascular disease in the developing world and its cost-effective management," *Circulation*, vol. 112, no. 23, pp. 3547–3553, 2005, doi: 10.1161/CIRCULATIONAHA.105.591792.

[27] S. S. Yadav and S. M. Jadhav, "Detection of common risk factors for diagnosis of cardiac arrhythmia using machine learning algorithm," *Expert Syst. Appl.*, vol. 163, no. July 2020, p. 113807, 2021, doi: 10.1016/j.eswa.2020.113807.

[28] G. Sannino and G. De Pietro, "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection," *Futur. Gener. Comput. Syst.*, vol. 86, pp. 446–455, 2018, doi: 10.1016/j.future.2018.03.057.

[29] E. J. da S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti, "ECG-based heartbeat classification for arrhythmia detection: A survey," *Comput. Methods Programs Biomed.*, vol. 127, pp. 144–164, 2016, doi: 10.1016/j.cmpb.2015.12.008.

[30] M. H. Wu and E. Y. Chang, "DeepQ arrhythmia database: A large-scale dataset for arrhythmia detector evaluation," *MMHealth 2017 - Proc. 2nd Int. Work. Multimed. Pers. Heal. Heal. Care, co-located with MM 2017*, pp. 77–80, 2017, doi: 10.1145/3132635.3132647.

[31] T. Tabassum and M. Islam, “An approach of cardiac disease prediction by analyzing ECG signal,” *2016 3rd Int. Conf. Electr. Eng. Inf. Commun. Technol. iCEEiCT 2016*, no. Mi, 2017, doi: 10.1109/CEEICT.2016.7873093.

[32] M. Tziakouri, C. Pitris, and C. Orphanidou, “Classification of AF and other arrhythmias from a short segment of ECG using dynamic time warping,” *Comput. Cardiol. (2010).*, vol. 44, pp. 1–4, 2017, doi: 10.22489/CinC.2017.348-295.

[33] A. Garfinkel *et al.*, “Preventing ventricular fibrillation by flattening cardiac restitution,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 97, no. 11, pp. 6061–6066, 2000, doi: 10.1073/pnas.090492697.

[34] K. Words, “V ENTRICULAR F IBRILLATION : Mechanisms of Initiation and Maintenance,” *New York*, pp. 25–50, 2000.

[35] Z. Ebrahimi, M. Loni, M. Daneshtalab, and A. Gharehbaghi, “A review on deep learning methods for ECG arrhythmia classification,” *Expert Syst. with Appl. X*, vol. 7, p. 100033, 2020, doi: 10.1016/j.eswax.2020.100033.

[36] B. E. Bussink, A. G. Holst, L. Jespersen, J. W. Deckers, G. B. Jensen, and E. Prescott, “Right bundle branch block: Prevalence, risk factors, and outcome in the general population: Results from the Copenhagen City Heart Study,” *Eur. Heart J.*, vol. 34, no. 2, pp. 138–146, 2013, doi: 10.1093/eurheartj/ehs291.

[37] M. Miszczyk *et al.*, “Clinical Evidence behind Stereotactic Radiotherapy for the Treatment of Ventricular Tachycardia (STAR)—A Comprehensive Review,” *J. Clin. Med.*, vol. 10, no. 6, p. 1238, 2021, doi: 10.3390/jcm10061238.

[38] R. M. John *et al.*, “Ventricular arrhythmias and sudden cardiac death,” *Lancet*, vol. 380, no. 9852, pp. 1520–1529, 2012, doi: 10.1016/S0140-6736(12)61413-5.

[39] S. Ayub and J. Saini, “ECG classification and abnormality detection using cascade forward neural network,” *Int. J. Eng. Sci. Technol.*, vol. 3, no. 3, pp. 41–46, 2011, doi: 10.4314/ijest.v3i3.68420.

[40] V. Krasteva and I. Jekova, “QRS template matching for recognition of ventricular ectopic beats,” *Ann. Biomed. Eng.*, vol. 35, no. 12, pp. 2065–2076, 2007, doi: 10.1007/s10439-007-9368-9.

[41] S. Sahoo, M. Dash, S. Behera, and S. Sabut, “Machine Learning Approach to Detect Cardiac Arrhythmias in ECG Signals: A Survey,” *Irbm*, vol. 41, no. 4, pp. 185–194, 2020, doi: 10.1016/j.irbm.2019.12.001.

[42] Y. Li and W. Cui, “Identifying the mislabeled training samples of ECG signals using machine learning,” *Biomed. Signal Process. Control*, vol. 47, pp. 168–176, 2019, doi: 10.1016/j.bspc.2018.08.026.

[43] J. Ferretti, V. Randazzo, G. Cirrincione, and E. Pasero, “1-D Convolutional Neural Network for ECG Arrhythmia Classification,” *Smart Innov. Syst. Technol.*, vol. 184, no. November 2020, pp. 269–279, 2021, doi: 10.1007/978-981-15-5093-5_25.

[44] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati, “Classification of ECG signals using machine learning techniques: A survey,” *Conf. Proceeding - 2015 Int. Conf. Adv. Comput. Eng. Appl. ICACEA 2015*, pp. 714–721, 2015, doi: 10.1109/ICACEA.2015.7164783.

[45] R. Barbieri, E. C. Matten, A. R. A. Alabi, and E. N. Brown, “A point-process model of human heartbeat intervals: New definitions of heart rate and heart rate variability,” *Am. J. Physiol. - Hear. Circ. Physiol.*, vol. 288, no. 1 57-1, pp. 424–435, 2005, doi: 10.1152/ajpheart.00482.2003.

[46] M. Abdi, A. Karimi, M. Navidbakhsh, G. Pirzad Jahromi, and K. Hassani, “A lumped parameter mathematical model to analyze the effects of tachycardia and bradycardia on the cardiovascular system,”

Int. J. Numer. Model. Electron. Networks, Devices Fields, vol. 28, no. 3, pp. 346–357, 2015, doi: 10.1002/jnm.2010.

[47] D. DeMazumder and G. F. Tomaselli, "Molecular and cellular mechanisms of cardiac arrhythmias," *Muscle*, vol. 1, pp. 583–599, 2012, doi: 10.1016/B978-0-12-381510-1.00041-7.

[48] S. Duffner, "Face Image Analysis With Convolutional Neural Networks," *Thesis*, p. 191, 2007.

[49] J. Bell, "Chapter 5 - Artificial Neural Networks," *Mach. Learn. Hands-On Dev. Tech. Prof.*, pp. 91–116, 2014.

[50] L. Zhang, "Implementation of Fixed-point Neuron Models with Threshold, Ramp and Sigmoid Activation Functions," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 224, no. 1, pp. 0–5, 2017, doi: 10.1088/1757-899X/224/1/012054.

[51] G. Asadollahfardi, *Artificial Neural Network*. 2015.

[52] "Neural Network Primer," no. 2, 2000.

[53] R. Mendes, P. Cortez, M. Rocha, and J. Neves, "Particle swarms for feedforward neural network training," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2, pp. 1895–1899, 2002, doi: 10.1109/ijcnn.2002.1007808.

[54] N. Kriegeskorte and T. Golan, "Neural network models and deep learning," *Curr. Biol.*, vol. 29, no. 7, pp. R231–R236, 2019, doi: 10.1016/j.cub.2019.02.034.

[55] S. Hikmat, A. Mohsin Abdulazeez, H. Saad, A. Haji, and A. Mohsin, "COMPARISON OF OPTIMIZATION TECHNIQUES BASED ON GRADIENT DESCENT ALGORITHM: A REVIEW PJAEE, 18 (4) (2021) COMPARISON OF OPTIMIZATION TECHNIQUES BASED ON GRADIENT DESCENT ALGORITHM: A REVIEW Comparison Of Optimization Techniques Based On Gradient Descent Al," vol. 18,

no. 4, pp. 2715–2743, 2021.

[56] S. Ruder, “An overview of gradient descent optimization algorithms,” pp. 1–14, 2016, [Online]. Available: http://arxiv.org/abs/1609.04747.

[57] I. Kandel, M. Castelli, and A. Popovič, “Comparative Study of First Order Optimizers for Image Classification Using Convolutional Neural Networks on Histopathology Images,” *J. Imaging*, vol. 6, no. 9, 2020, doi: 10.3390/JIMAGING6090092.

[58] J. Zhang, “Gradient Descent based Optimization Algorithms for Deep Learning Models Training,” 2019, [Online]. Available: http://arxiv.org/abs/1903.03614.

[59] R. Atienza, *Advanced Deep Learning with Keras*. 2018.

[60] G. Çınarer, B. G. Emiroğlu, R. S. Arslan, and A. H. Yurttakal, *Brain tumor classification using deep neural network*, vol. 5, no. 5. Springer Singapore, 2020.

[61] R. Chauhan, K. K. Ghanshala, and R. C. Joshi, “Convolutional Neural Network (CNN) for Image Detection and Recognition,” *ICSCCC 2018 - 1st Int. Conf. Secur. Cyber Comput. Commun.*, pp. 278–282, 2018, doi: 10.1109/ICSCCC.2018.8703316.

[62] R. J. Erb, “Introduction to Backpropagation Neural Network Computation,” *Pharmaceutical Research: An Official Journal of the American Association of Pharmaceutical Scientists*, vol. 10, no. 2. pp. 165–170, 1993, doi: 10.1023/A:1018966222807.

[63] J. Li, J. H. Cheng, J. Y. Shi, and F. Huang, “Brief introduction of back propagation (BP) neural network algorithm and its improvement,” *Adv. Intell. Soft Comput.*, vol. 169 AISC, no. VOL. 2, pp. 553–558, 2012, doi: 10.1007/978-3-642-30223-7_87.

[64] A. Wanto, A. P. Windarto, D. Hartama, and I. Parlina, “Use of Binary Sigmoid Function And Linear Identity In Artificial Neural Networks For

Forecasting Population Density," *IJISTECH (International J. Inf. Syst. Technol.*, vol. 1, no. 1, p. 43, 2017, doi: 10.30645/ijistech.v1i1.6.

[65] H. K. Jabbar and R. Z. Khan, "Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning (Comparative Study)," no. February, pp. 163–172, 2015, doi: 10.3850/978-981-09-5247-1_017.

[66] X. Ying, "An Overview of Overfitting and its Solutions," *J. Phys. Conf. Ser.*, vol. 1168, no. 2, 2019, doi: 10.1088/1742-6596/1168/2/022022.

[67] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, and D. Batra, "Reducing overfitting in deep networks by decorrelating representations," *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, pp. 1–12, 2016.

[68] D. Lee *et al.*, "Regularization Strategy for Point Cloud via Rigidly Mixed Sample," 2021, [Online]. Available: http://arxiv.org/abs/2102.01929.

[69] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Comput.*, vol. 22, no. 1, 2018, doi: 10.1007/s00500-016-2442-1.

[70] Y. A. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, *Efficient backprop*, vol. 7700 LECTU, no. 1998. 2012.

[71] N. Hatami, Y. Gavet, and J. Debayle, "using deep convolutional neural networks," no. April 2018, 2021, doi: 10.1117/12.2309486.

[72] R. Andonie and A. C. Florea, "Weighted random search for CNN hyperparameter optimization," *Int. J. Comput. Commun. Control*, vol. 15, no. 2, pp. 1–11, 2020, doi: 10.15837/IJCCC.2020.2.3868.

[73] T. M. Breuel, "The Effects of Hyperparameters on SGD Training of Neural Networks," 2015, [Online]. Available: http://arxiv.org/abs/1508.02788.

[74] L. N. Smith, “Cyclical Learning Rates for Training Neural Networks,” no. April, 2015.

[75] J. Brownlee, “What is the Difference Between a Batch and an Epoch in a Neural Network?,” *Mach. Learn. Mastery*, no. July, pp. 3–4, 2018, [Online]. Available: https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/.

[76] T. Hinz, N. Navarro-Guerrero, S. Magg, and S. Wermter, “Speeding up the Hyperparameter Optimization of Deep Convolutional Neural Networks,” *Int. J. Comput. Intell. Appl.*, vol. 17, no. 2, pp. 1–15, 2018, doi: 10.1142/S1469026818500086.

[77] I. Loshchilov and F. Hutter, “CMA-ES for Hyperparameter Optimization of Deep Neural Networks,” no. 2001, 2016, [Online]. Available: http://arxiv.org/abs/1604.07269.

[78] X. Xiao, M. Yan, S. Basodi, C. Ji, and Y. Pan, “Efficient Hyperparameter Optimization in Deep Learning Using a Variable Length Genetic Algorithm,” 2020, [Online]. Available: http://arxiv.org/abs/2006.12703.

[79] J. Feng and S. Lu, “Performance Analysis of Various Activation Functions in Artificial Neural Networks,” *J. Phys. Conf. Ser.*, vol. 1237, no. 2, 2019, doi: 10.1088/1742-6596/1237/2/022030.

[80] S. Sharma and S. Sharma, “Understanding Activation Functions in Neural Networks,” *Int. J. Eng. Appl. Sci. Technol.*, vol. 4, no. 12, pp. 310–316, 2017.

[81] C. E. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation Functions : Comparison of Trends in Practice and Research for Deep Learning,” pp. 1–20.

[82] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, “Learning activation functions to improve deep neural networks,” *3rd Int. Conf.*

Learn. Represent. ICLR 2015 - Work. Track Proc., no. 2013, pp. 1–9, 2015.

[83] S. Kanai, Y. Yamanaka, Y. Fujiwara, and S. Adachi, "SigsofTmax: Reanalysis of the softmax bottleneck," *Adv. Neural Inf. Process. Syst.*, vol. 2018-Decem, pp. 286–296, 2018.

[84] I. Kouretas and V. Paliouras, "Simplified Hardware Implementation of the Softmax Activation Function," *2019 8th Int. Conf. Mod. Circuits Syst. Technol. MOCAST 2019*, pp. 1–4, 2019, doi: 10.1109/MOCAST.2019.8741677.

[85] D. Pedamonti, "Comparison of non-linear activation functions for deep neural networks on MNIST classification task," no. 3, 2018, [Online]. Available: http://arxiv.org/abs/1804.02763.

[86] H. H. Chieng, N. Wahid, and P. Ong, "Parametric Flatten-T Swish: an Adaptive Nonlinear Activation Function for Deep Learning," *J. Inf. Commun. Technol.*, vol. 20, no. 1, pp. 21–39, 2021, doi: 10.32890/jict.20.1.2021.9267.

[87] B. Zoph and Q. V Le, "Searching for activation functions," *6th Int. Conf. Learn. Represent. ICLR 2018 - Work. Track Proc.*, pp. 1–13, 2018.

[88] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Comput. Sci. Rev.*, vol. 40, p. 100379, 2021, doi: 10.1016/j.cosrev.2021.100379.

[89] B. Lim and S. Zohren, "Time-series forecasting with deep learning: A survey," *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 379, no. 2194, 2021, doi: 10.1098/rsta.2020.0209.

[90] H. Mohsen, E.-S. A. El-Dahshan, E.-S. M. El-Horbaty, and A.-B. M. Salem, "Classification using deep learning neural networks for brain tumors," *Futur. Comput. Informatics J.*, vol. 3, no. 1, pp. 68–71, 2018, doi: 10.1016/j.fcij.2017.12.001.

[91] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” *Comput. Electron. Agric.*, vol. 147, no. July 2017, pp. 70–90, 2018, doi: 10.1016/j.compag.2018.02.016.

[92] M. Salem, S. Taheri, and J. Yuan, “ECG Arrhythmia Classification Using Transfer Learning from 2-Dimensional Deep CNN Features,” *2018 IEEE Biomed. Circuits Syst. Conf.*, pp. 1–4, 2018.

[93] C. Rao and Y. Liu, “Three-dimensional convolutional neural network (3D-CNN) for heterogeneous material homogenization,” *Comput. Mater. Sci.*, vol. 184, pp. 1–21, 2020, doi: 10.1016/j.commatsci.2020.109850.

[94] S. Rani and P. Kumar, “Deep Learning Based Sentiment Analysis Using Convolution Neural Network,” *Arab. J. Sci. Eng.*, vol. 44, no. 4, pp. 3305–3314, 2019, doi: 10.1007/s13369-018-3500-z.

[95] W. Y. Lee, S. M. Park, and K. B. Sim, “Optimal hyperparameter tuning of convolutional neural networks based on the parameter-setting-free harmony search algorithm,” *Optik (Stuttg).*, vol. 172, no. May, pp. 359–367, 2018, doi: 10.1016/j.ijleo.2018.07.044.

[96] C. Chen, Z. Hua, R. Zhang, G. Liu, and W. Wen, “Automated arrhythmia classification based on a combination network of CNN and LSTM,” *Biomed. Signal Process. Control*, vol. 57, p. 101819, 2020, doi: 10.1016/j.bspc.2019.101819.

[97] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–22, 2021, doi: 10.1109/TPAMI.2021.3059968.

[98] M. A. Haroon, “ECG Arrhythmia classification using Deep Convolution Neural Networks in Transfer Learning,” no. June, 2020.

[99] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M.

Adam, “Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals,” *Inf. Sci. (Ny).*, vol. 415–416, pp. 190–198, 2017, doi: 10.1016/j.ins.2017.06.027.

[100] O. Abdel-Hamid, L. Deng, and D. Yu, “Exploring convolutional neural network structures and optimization techniques for speech recognition,” *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*, no. August, pp. 3366–3370, 2013.

[101] “School of Computing Master ’ s Thesis A Deep Learning (Neural Network) Approach for ECG Heartbeat Classification : Arrhythmia Detection Bukola Salami,” no. January, 2020.

[102] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks,” pp. 1–11, 2015, [Online]. Available: http://arxiv.org/abs/1511.08458.

[103] Z. Yi, “Evaluation and Implementation of Convolutional Neural Networks in Image Recognition,” *J. Phys. Conf. Ser.*, vol. 1087, no. 6, 2018, doi: 10.1088/1742-6596/1087/6/062018.

[104] P. Singh, P. Raj, and V. P. Namboodiri, “EDS pooling layer,” *Image Vis. Comput.*, vol. 98, p. 103923, 2020, doi: 10.1016/j.imavis.2020.103923.

[105] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/ICEngTechnol.2017.8308186.

[106] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020, doi: 10.1007/s10462-020-09825-6.

[107] W. L. Mao, W. C. Chen, C. T. Wang, and Y. H. Lin, “Recycling waste

classification using optimized convolutional neural network," *Resour. Conserv. Recycl.*, vol. 164, no. August 2020, p. 105132, 2021, doi: 10.1016/j.resconrec.2020.105132.

[108] W. Ma and J. Lu, "An Equivalence of Fully Connected Layer and Convolutional Layer," no. 3, pp. 1–9, 2017, [Online]. Available: http://arxiv.org/abs/1712.01252.

[109] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[110] F. Wu, P. Hu, and D. Kong, "Flip-Rotate-Pooling Convolution and Split Dropout on Convolution Neural Networks for Image Classification," 2015, [Online]. Available: http://arxiv.org/abs/1507.08754.

[111] J. Jin, A. Dundar, and E. Culurciello, "Flattened convolutional neural networks for feedforward acceleration," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Work. Track Proc.*, no. 2014, pp. 1–11, 2015.

[112] G. S. Nandini, A. P. S. Kumar, and C. K, "Dropout technique for image classification based on extreme learning machine," *Glob. Transitions Proc.*, vol. 2, no. 1, pp. 111–116, 2021, doi: 10.1016/j.gltp.2021.01.015.

[113] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mech. Syst. Signal Process.*, vol. 151, p. 107398, 2021, doi: 10.1016/j.ymssp.2020.107398.

[114] W. Tang, G. Long, L. Liu, T. Zhou, J. Jiang, and M. Blumenstein, "Rethinking 1D-CNN for Time Series Classification: A Stronger Baseline," 2020, [Online]. Available: http://arxiv.org/abs/2002.10061.

[115] H. M and S. M.N, "A Review on Evaluation Metrics for Data Classification Evaluations," *Int. J. Data Min. Knowl. Manag. Process*, vol. 5, no. 2, pp. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.

[116] S. Kaplan Berkaya, A. K. Uysal, E. Sora Gunal, S. Ergin, S. Gunal, and M. B. Gulmezoglu, “A survey on ECG analysis,” *Biomed. Signal Process. Control*, vol. 43, pp. 216–235, 2018, doi: 10.1016/j.bspc.2018.03.003.

[117] A. Gunawardana and G. Shani, “A survey of accuracy evaluation metrics of recommendation tasks,” *J. Mach. Learn. Res.*, vol. 10, pp. 2935–2962, 2009.

[118] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, “What Do Different Evaluation Metrics Tell Us about Saliency Models?,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 740–757, 2019, doi: 10.1109/TPAMI.2018.2815601.

[119] C. Marzban, “The ROC curve and the area under it as performance measures,” *Weather Forecast.*, vol. 19, no. 6, pp. 1106–1114, 2004, doi: 10.1175/825.1.

[120] Sarang Narkhede, “Understanding AUC - ROC Curve,” *Towar. Data Sci.*, pp. 6–11, 2019, [Online]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.

[121] J. Pestana, D. Belo, and H. Gamboa, “Detection of abnormalities in electrocardiogram (ECG) using deep learning,” *BIOSIGNALS 2020 - 13th Int. Conf. Bio-Inspired Syst. Signal Process. Proceedings; Part 13th Int. Jt. Conf. Biomed. Eng. Syst. Technol. BIOSTEC 2020*, pp. 236–243, 2020, doi: 10.5220/0008967302360243.

[122] A. E. Maxwell, T. A. Warner, and L. A. Guillén, “Accuracy assessment in convolutional neural network-based deep learning remote sensing studies—part 2: Recommendations and best practices,” *Remote Sens.*, vol. 13, no. 13, 2021, doi: 10.3390/rs13132591.

[123] T. Jayalakshmi and A. Santhakumaran, “Statistical Normalization and Back Propagationfor Classification,” *Int. J. Comput. Theory Eng.*, vol.

3, no. 1, pp. 89–93, 2011, doi: 10.7763/ijcte.2011.v3.288.

[124] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Appl. Soft Comput.*, vol. 97, p. 105524, 2020, doi: 10.1016/j.asoc.2019.105524.

المستخلص

تعد أمراض القلب والأوعية الدموية، بما في ذلك السكتات الدماغية وأمراض القلب، نوعًا بارزًا من الأمراض المهددة للحياة والأمراض المزمنة، وأحد الأسباب الرئيسية للوفيات في جميع أنحاء العالم. أحد أكثر الأمراض القلبية الوعائية شيوعًا هو عدم انتظام ضربات القلب، وهو حالة قلبية تحدث نتيجة خلل في ضربات القلب، مما يعني ان الإشارات الكهربائية للقلب لا تعمل بشكل صحيح، مما يؤدي الى عدم انتظام ضربات القلب أو النظم وبالتالي عجز ضخ الدم. في الروتين السريري، يتم الكشف عن عدم انتظام ضربات القلب عن طريق إشارات تخطيط القلب الكهربائي وهي أداة غير جراحية تستخدم لتشخيص أمراض القلب، وخاصة عدم انتظام ضربات القلب من خلال إشاراتها التي يمكن ان تكشف عن نشاط غير طبيعي للقلب.

يعتبر تشخيص عدم انتظام ضربات القلب في الروتين الطبي من خلال تخطيط القلب الكهربائي عن طريق التحقق من النبض، إجراءً شاقًا ويستغرق وقتًا طويلاً، بالإضافة إلى صعوبة التفسير البصري لإشارات تخطيط القلب الكهربائي نظرًا لصغر اتساعها ومدتها. لذلك، يمكن أن يساعد التشخيص التلقائي لعدم انتظام ضربات القلب بواسطة الخوارزمية المحوسبة في استخدام الأجهزة القابلة للارتداء كنظم مراقبة صحية يومية، حيث قد يؤدي تقليل تعقيد الحوسبة وزيادة الدقة إلى التشخيص الصحيح. تم اقتراح المزيد من الأعمال، والتي جاءت من التعلم الآلي والتعلم العميق من أجل تحقيق أداء عالٍ. ومع ذلك، هنالك تحد للوصول الى هذا الهدف.

نتيجة لذلك، قدمنا نهجًا مهمًا لتحديد عدم انتظام ضربات القلب باستخدام إشارات تخطيط القلب. حيث اقترحنا في هذه الدراسة نهجًا يعتمد على تقنية التعلم العميق الذي يمثل إطارًا لشبكة عصبية تلافيفيه ذات بُعد واحد من تسعة طبقات لتصنيف إشارات تخطيط القلب الكهربائي تلقائيًا إلى أمراض قلبية مختلفة باستخدام تصنيفات مختلفة كتصنيف ثنائي ومتعدد الفئات.

في التصنيف الثنائي، يكشف نموذج الشبكة العصبية التلافيفية المقترح الفئة الطبيعية من الغير طبيعية بينما في التصنيف المتعدد الفئات سبعه أنواع مختلفة من أشارات تخطيط

القلب الكهربائي والتي تسمى الطبيعي، حصار الغصن الأيسر، حصار الغصن الأيمن، الضرب الاذيني المبكر، انقباض البطين المبكر، الضربة المندمجة، والنبض السريع يتم تشخيصها باستخدام تصنيفات مختلفة مثل تصنيف ثلاث ،اربع ،ست وسبع اصناف.

تم اجراء اختبار عملي لهذا العمل باستخدام قاعدة بيانات MIT-BIH القياسية. وقد حقق النموذج المقترح اعلى النتائج في تصنيف ثلاث اصناف والذي يصنف ثلاث أنواع مختلفه من عدم انتظام ضربات القلب وهي الضرب الاذيني المبكر، حصار الغصن الأيسر، وحصار الغصن الأيمن بمتوسط ضبط 99.52%، دقة 99.34%، استرجاع 99.26%، وخصوصيه 99.76%، ودرجه 99.30% في وقت تدريب 361 ثانيه.

في النهاية تمت مقارنة النظام المقترح مع بعض النماذج الحديثة ذات الصلة، واظهرت النتائج المحققة تفوق الإطار المقترح على تلك النماذج في معظم معايير التقييم, مما يدل على نجاح النهج الجديد.

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة القادسية
كليه علوم الحاسوب وتكنلوجيا المعلومات

منهج مقترح لتصنيف إشارة مخطط كهربائية القلب باستخدام الشبكة العصبية التلافيفية أحادية البعد

رسالة

مقدمه إلى مجلس كلية علوم الحاسوب وتكنلوجيا المعلومات بجامعة القادسية في استيفاء جزئي لمتطلبات درجة الماجستير في علوم الحاسوب

من قبل

ساره كامل گطفان

بإشراف

أ.م.د. لمياء عبدنور محمد

١٤٤٣هـ ٢٠٢١ م