Micro Classes ECG Classification using Convolutional Neural Network

A project report submitted in fulfillment of the requirements for B.Tech. Project

Integrated Post Graduate (M.Tech)

by

Vishal Verma (2019IMT-116)

Under supervison of
Prof. Manisha Pattanaik

विश्वजीवनामृतं ज्ञानम्

ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT GWALIOR-474 015

2022

CANDIDATES DECLARATION

I hereby certify that the work, which is being presented in the report, entitled **Micro Classes ECG Classification using Convolutional Neural Network**, in fulfillment of the requirement for the award of the Degree of **Bachelor of Technology** and submitted to the institution is an authentic record of my own work carried out during the period *June 2022* to *September 2022* under the supervision of **Prof. Manisha Pattanaik**. I have also cited the references about the text(s)/figure(s)/table(s) from where they have been taken.

Date: Signatures of the Candidates

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date: Signatures of the Research Supervisor

ABSTRACT

The leading cause of death in this day and time is cardiovascular disease (CVD). Cardiovascular abnormalities are usually detected by using noninvasive electrocardiogram (ECG). Information on the electrophysiology of cardiac disorders and potential ischemia changes is provided by a cleaned ECG signal. It offers insightful knowledge about various arrhythmia types, the cardiovascular system, and the heart's functionality. Manual analysis of (ECG) data from Holter monitors is quite difficult. As a corollary, each heartbeat should be automatically identified and classified using a computer-aided process. Convolutional neural networks (CNNs) have drawn a lot of attention because they can automatically classify ECG signals without laborious manual feature extractions. However, there are some drawbacks to these typical methods, requiring manual feature recognition, complex models, and long training time and more trainable parameters with high computational cost. Due to high computational cost existing models are not very useful in edge computing. The main objective of this project is to reduce the number of trainable parameters and thereby reducing model complexity. In this project, we proposed a modified 12-layer deep one dimensional convolutional neural network, compared to 12 layer deep one dimensional convolutional neural network, on classifying the five micro-classes of heartbeat types in the MIT- BIH Arrhythmia database. The five types of heartbeat features are classified, and a wavelet transform denoising method is used to denoise the raw ECG signal. Compared with 12-layer deep 1D convolutional neural network, the modified 12-layer deep 1D convolutional neural network significantly reduced the trainable parameters. The trainable parameters are reduced from 292,887 to 106,997.

*Keywords:*ECG, Micro-classes, complexity, MIT-BIH, CNN, Denoise , 12-layer deep 1D CNN, wavelet.

ACKNOWLEDGEMENTS

I am extremely grateful to Prof. Manisha Pattanaik for giving me the freedom to work on and experiment with new ideas. I'd like to take this chance to show my sincere gratitude to them for their academic advice, as well as for their interest in and ongoing support of my project, as well as for the sessions they held to raise my confidence and motivate me. These sessions proved to be very helpful and were crucial in helping me develop a sense of self-assurance and trust. They offered priceless suggestions, recommendations, wise judgement, constructive criticism, and a perfectionism that helped the current work advance and thrive.

Finally, I wish to express my gratitude to Mr. Anil Kumar Rajput for his steadfast direction of my project. I'd want to conclude by thanking my institution and my friends, whose unceasing support lifted my spirits, helped me refocus my efforts, and made it possible for me to complete this work.

{Vishal Verma}

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

CNN	Convolution neural network
ECG	Electrocardiogram
DNN	Deep neural network
NOR	Normal
LBBB	Left Bundle Branch Block
RBBB	Right Bundle Branch Block
AP	Atrial Premature
PVC	Premature Ventricular Contraction
EEG	Electroencephalogram
EMG	Electromyogram
1D	One Dimensional
MIT-BIH	Massachusetts Institute of Technology-Beth Israel Hospital
AI	Artificial Intelligence

CHAPTER 1

Introduction

1.1 General Overview

The population is growing day after day. People therefore don't benefit sufficiently from health care services. It is impossible to keep patients under a doctor's care at all times. For medical practitioners, time and speed are crucial. This project is intended to be used with edge computing devices(i.e devices with less computation), like in smart watches. This Project can also be used for education.

1.2 Context

The health of people, especially those in their middle years and older, is seriously threatened by cardiovascular disease, which is a frequent condition. High prevalence, high disability, and high death are its defining traits. The population of the world is ageing currently. Cardiovascular disease's worsening progression has grown to be a major public health concern [1]. Analyzing an ECG is a useful tool for assessing heart health [2]. As a result, understanding and classifying ECG signals is essential for comprehending cardiovascular disorders. Not just for early detection and appropriate treatment, but also for early prevention. Examining the categorization of linked ECG signals is extremely important [3] [4].

1.3 The Electrocardiogram

Doctors treated internal organ disorders with open and examine techniques. It implied that agony persisted continuously. Both doctors and patients found this scenario to be exceedingly challenging. Modern technology is better. Doctors now employ non-invasive techniques. for instance, blood and lab tests. Different signals are obtained from the human body's organs. These signals produce information that belongs to

this group. EMG (electrical activity of the muscles, electromyogram), EEG (electrical activity of the brain, electroencephalogram), and ECG are a few examples. In this project, we carried out ECG signal analysis. Below is further information on this signal. When the heart experiences depolarization (the electrical activation of the tissues is positive) and repolarization, the electrical current is distributed throughout the body (The discharge of electrical charge of the tissues). The heart is what causes this electrical activity. An array of electrodes positioned on the surface of the body can be used to measure it. Electrocardiograms are the name given to these data (ECG or EKG).
The use of microcomputers to process the electrocardiogram (ECG) has become more popular during the past several years. According to a review of the literature in this field, microcomputer-based systems are remarkably capable of providing necessary medical services.

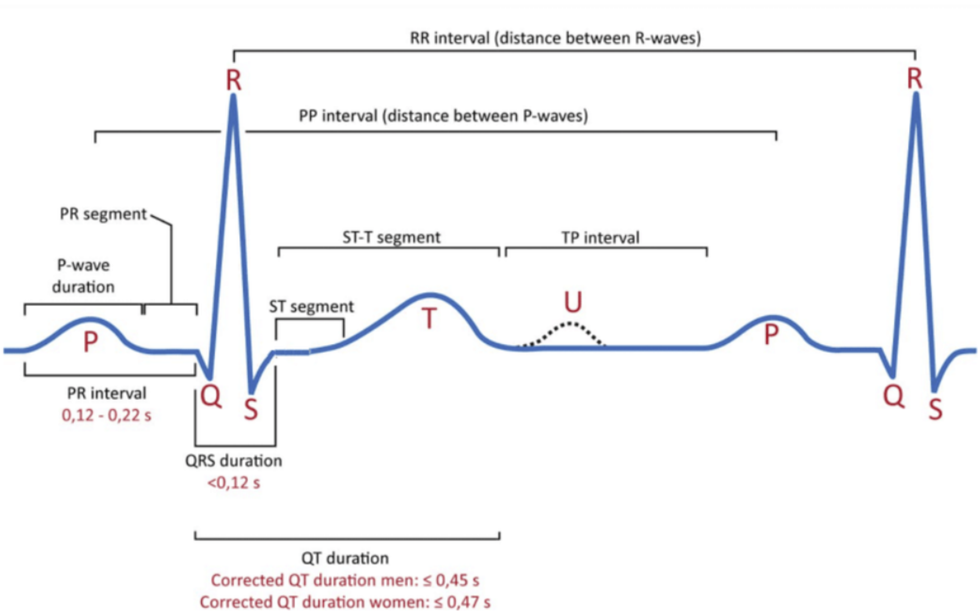

Figure 1.1: Classical ECG signal characteristics

Figure 1.1 analyzes the heart beat period of time curve (typical ECG waveforms). The figure has a crispness to it. In Figure 1.1, periodic sharpness may be observed. The ventricular contraction is associated to this sharpness. They are termed as the "depolarization." During depolarization, muscle fibres loose their resting potential. Signals that are often about to take a minor negative deviation continue with a significant positive taper and then a second negative deviation income. The "QRS complex" refers to the ventricles’ contraction. After the QRS complex, an oscillation is noticed. The ST range or ST wave are other names for this oscillation. The potential difference at this point is incalculable. At this point, an isoelectric line, also known as a zero line, is drawn. T-wave follows after this (representing the depolarization). The P wave suggests that they are in the excited state to the atrium. Finally, a U wave is not constantly visible.

Typically, it is little. The repolarization of the papillary muscles is assumed to be represented by U waves. In Figure 1.2, an ECG signal is shown. (This signal is purely hypothetical. As compared to this signal, the genuine ECG signal is less smooth.)

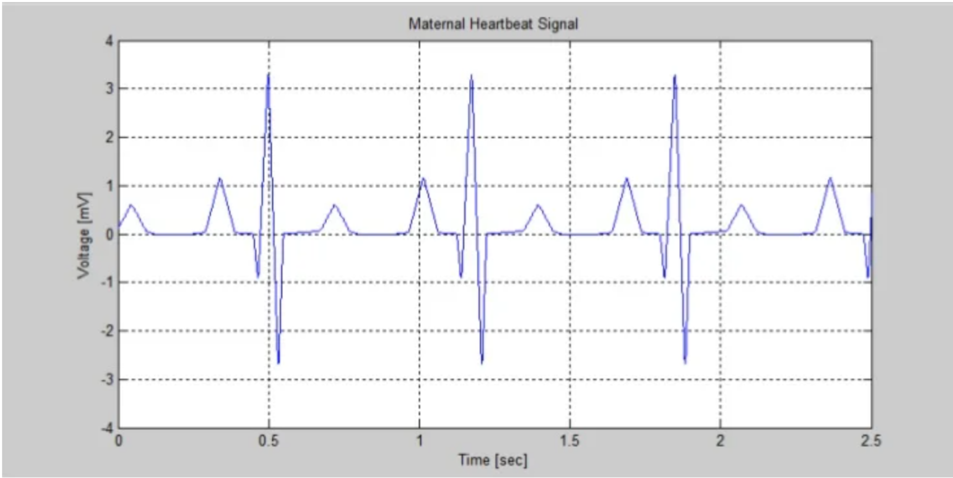

Figure 1.2: Shape of electrocardiogram

An ECG, which can be collected from the skin's surface and from various angles like in Figure 1.3, is a evaluation of the electrical impulses of the heart muscle. Action potentials are released mechanically inside the heart muscle during contractions that pump blood to all areas of the body. This mechanical process results in electrical activity.

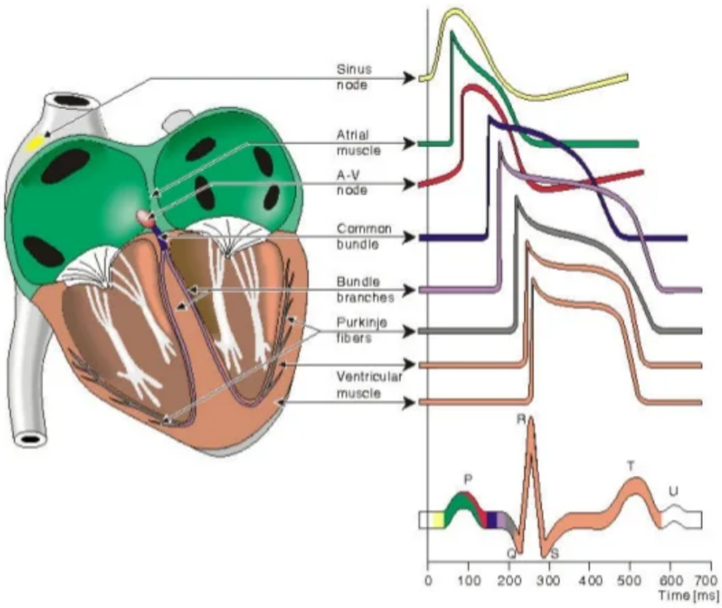

Figure 1.3: The heart's anatomy with waveforms from different specific part of the heart

1.4 ECG Waveform Description

The ECG wave is created as a prediction of the heart's potential vectors that have been condensed. The ECG wave features a number of peaks and "formations," which is helpful for diagnosing the wave.These are:

1. P-wave - shows the depolarized wave that propagates from the SA node to the atria, and its time constant is 80–100 ms.

2. P-R interval -suggests that the time required for an electrical impulse to go from the sinus node to AV node before accessing the ventricles is 120 to 200 milliseconds.

3. P-R segment - Corresponds to the time between the ends of atrial depolarization to the onset of ventricular depolarization. Last about 100ms.

4. QRS complex - Represents ventricular depolarization. The duration of the QRS complex is normally 0.06 to 0.1 seconds.

5. Q-wave - Represents the normal left-to-right depolarization of the inter ventricular septum.

6. R-wave - Represents early depolarization of the ventricles.

7. S-wave - Represents late depolarization of the ventricles.

8. S-T segment — it appears after QRS and indicates that the entire ventricle is depolarized.

9. Q-T interval - is an estimate of the typical ventricular action length since it shows the total time required for both ventricular repolarization and depolarization to occur. This period can change according to heart rate and range from 0.2 to 0.4 seconds.

10. T-wave - indicates that the period for ventricular repolarization is larger than depolarization.

1.5 Motivation

Heart disease can take many different forms, and time-consuming manual diagnosis makes it simple to produce false positives. A new issue is how to precisely and rapidly analyse particular diseases. Additionally, the unpredictable, low-frequency, and vulnerable characteristics of ECG signals cause the diagnosis results to be unstable. To increase the effectiveness and accuracy of ECG recognition, advanced automatic identification and classification of ECG signals has become a necessity [5]. We studied the five categories of micro-classification heartbeats: Normal (NOR), Left Bundle Branch Block (LBBB), Right Bundle Branch Block (RBBB), Atrial Premature (AP), and Premature Ventricular Contraction(PVC) because there is relatively little effort put into classifying the micro-classes of the ECG signal. Also there is very little effort devoted to reduce the compuational cost of the existing models to classify the heartbeats, so that it can be used for edge computing devices. Therefore, a more accurate but with less computational cost(i.e less trainable parameters) and low-cost diagnosis of arrhythmia heartbeats is desirable. In this project, firstly we used wavelet self adaptive threshold method to remove the noise from raw ecg signal, secondly we implemented a modified 12-layer deep one-dimensional convolutional neural network which reduces the trainable parameters compared to [6] on classifying the five micro-classes of heartbeat types in the MIT- BIH Arrhythmia database.

1.6 Report Layout

Chapter 2 Literature Review tells us about the background, key related researches done and help us to identify the gaps in existing literature. Our objectives are also described in this chapter.
Chapter 3 Methodology discusses the working of our project, graphical abstract, modified architecture, proposed hypothesis, about dataset ,pre-processing etc.
Chapter 4 Results helps us understand the results using graphs, classification reports, summary table of the model and various representation models.
Chapter 5 Conclusion tells the contributions,limitations and future scope of our project.

CHAPTER 2

Literature review

This section is about the research papers and articles referred for any information and knowledge regarding the project.

2.1 Background

With the development of AI-based technology, a variety of machine learning techniques are being employed in the identification of ECG signal features in an effort to address issues relating to the enormous amount of data on ECG signal features and the laborious nature of manual detection. Along with making outstanding advancements in the domains of image processing, audio identification, and many other areas in recent years [7] [8] [9] [10], machine learning and deep learning networks have also become widely employed in the assisted detection of heart illness based on ECG data [11] [12] [13].

Deep learning networks are able to recognise complicated data patterns, automatically extract features, and do away with complex signal preparation. Additionally, deep learning networks are better at recognising single-lead, multi-class, and imbalanced ECG datasets thanks to their higher nonlinear fitting capabilities [12]. The classification of arrhythmia ECG signals has been effectively accomplished using the convolutional neural network (CNN), a feedforward neural network that has been the subject of extensive research and use in deep learning [11] [12] [13].

2.2 Key Related Research

Xu, Xuexiang and Liu, Hongxing "ECG Heartbeat Classification Using Convolutional Neural Networks". This paper tells us about the problems related to manual classification of beats. Manual classification is extremely hard to analyze and also time consuming. Manual classification many times falsely detects arrhythmias. Therefore, it is necessary to automatically analyze and categorize each heartbeat using a computer-aid method. Hence, convolutional neural networks (CNNs) can classify ECG signals automatically without trivial manual feature extractions. [14]

Tao Wang, Changhua Lu, Yining Sun, Mei Yang, Chun Liu, Chunsheng Ou "Automatic ECG Classification Using Continuous Wavelet Transform and Convolutional Neural Network". The automatic ECG classification approach described in this paper uses Continuous Wavelet Transform (CWT) and Convolutional Neural Network (CNN). CNN is used to extract features from the 2D-scalogram made up of the various time-frequency components obtained by CWT's decomposition of the ECG signals. Four RR interval characteristics are retrieved and coupled with CNN features to be fed into the dense layer for ECG classification, taking into account that the surrounding R peak interval (also known as RR interval) is helpful for the detection of arrhythmia [15].

Mengze Wu, Yongdi Lu, Wenli Yang, Shen Yuong Wong "A Study on Arrhythmia via ECG Signal Classification Using the Convolutional Neural Network". This paper discusses several issues with some of the previous research, such as manual feature recognition, intricate models, and lengthy training times. It also indicates that little to no research has been done on the micro classes of the Electrocardiogram signal. In order to categorise the five micro-classes of heartbeat types in the MIT-BIH Arrhythmia database, this research offers a reliable and effective 12-layer deep one-dimensional convolutional neural network. The five different types of heartbeat characteristics are categorised, and the tests employ the wavelet self-adaptive threshold denoising method [6].

Min Lin, Qiang Chen, Shuicheng Yan "Network In Network". This paper tells us about using Global average pooling instead of fully connected layers in standard convolutional neural networks. As it enforces correlation among feature maps and categories, which is made feasible by a stronger local modelling utilising the micro network, global average pooling is more relevant and understandable. Additionally, the dense layers are vulnerable to overfitting and rely largely on dropout regularisation [16] [17], whereas global average pooling is a structural regularizer by itself and naturally guards against overfitting for the entire structure [18].

Sabrina Göllner "How to reduce training parameters in CNNs while keeping accuracy >99%". This article tell 3 approaches to reduce trainable parameters→ (a) By using MaxPooling which downsampled the input for next convolutional layer, (b) using global average pooling instead of fully connected layers, which subsequently decreased the trainable parameters, (c) By using pruning technique, where pruning of layers or pruning of filter and kernel size is done to reduce parameters.

Masko, David, Hensman, Paulina "The Impact of Imbalanced Training Data for Convolutional Neural Networks". This paper studies the impact of imbalanced training data on Convolutional Neural Network (CNN) performance in image classification. It is concluded that oversampling is a viable way to counter the impact of imbalances in the training data.

Rakshith Vasudev "Understanding and Calculating the number of Parameters in Convolution Neural Networks (CNNs)". This article tells us about how CNN learns using parameters. It also elaborately describes the contribution of input layer, CNN layer, Pool layer, Fully Connected layer, softmax layer to the trainable parameters.

Kizito Nyuytiymbiy "Parameters and Hyperparameters in Machine Learning and Deep Learning". This article tells us about the key difference between parameters and hyperparameters. In deep learning and machine learning, a hyperparameter is essentially anything whose values or configuration are chosen before training starts and whose values or settings will stay the same when training is complete.

2.3 Research Gaps

1. To increase the accuracy of model less, attention was given to the complexity of model and models are highly complex and needs more computation and training time.
2. Less attention was given towards classification of micro classes present in MIT-BIH arrhytmia dataset.
3. Less use of global average pooling, instead fully connected layers are used in ECG classification which leads to overfitting.
4. The previous model [6] was highly complex, hence not suitable for edge computing.

2.4 Objectives

The objective of the project are as follows:

1. To reduce the complexity of the 12-layer deep 1D CNN as a step towards edge computing.
2. To build a beat classification model using deep learning techniques(using CNN).
3. To denoise raw ECG signal using wavelet transform method.
4. To classify the five micro-classes, (normal (NOR), left bundle branch block (LBBB), right bundle branch block (RBBB), Atrial premature beats (AP), and premature ventricular beats (PVC), of heartbeat types in the MIT-BIH Arrhythmia database.

CHAPTER 3

Methodology

The methodology is a relevant structure for research. We have multiple sections that covers the Graphical abstract, Network architecture, proposed hypothesis, About dataset, pre-processing, Data segmentation, Data enhancement, Understanding parameters, hyperparameters and globalaveragepooling, mechanism, Implementation details.

3.1 Graphical Abstract

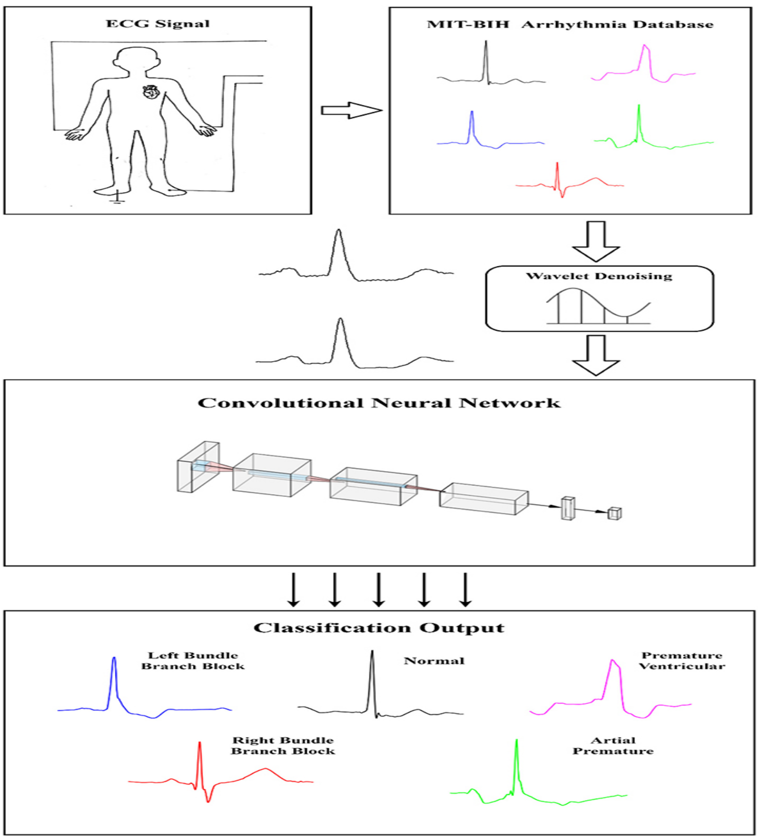

Figure 3.1: The graphical abstract of the methodology

3.2 Modified 12-Layer Deep 1D CNN Model

The modified architecture consists of three parts, Figure 3.2:

1. Raw ECG signal preprocessing, which includes signal denoising, data segmentation and then data enhancement
2. Then feeding the processed ECG signal to the proposed model
3. Finally classification output of five-micro heartbeats will be received at end.

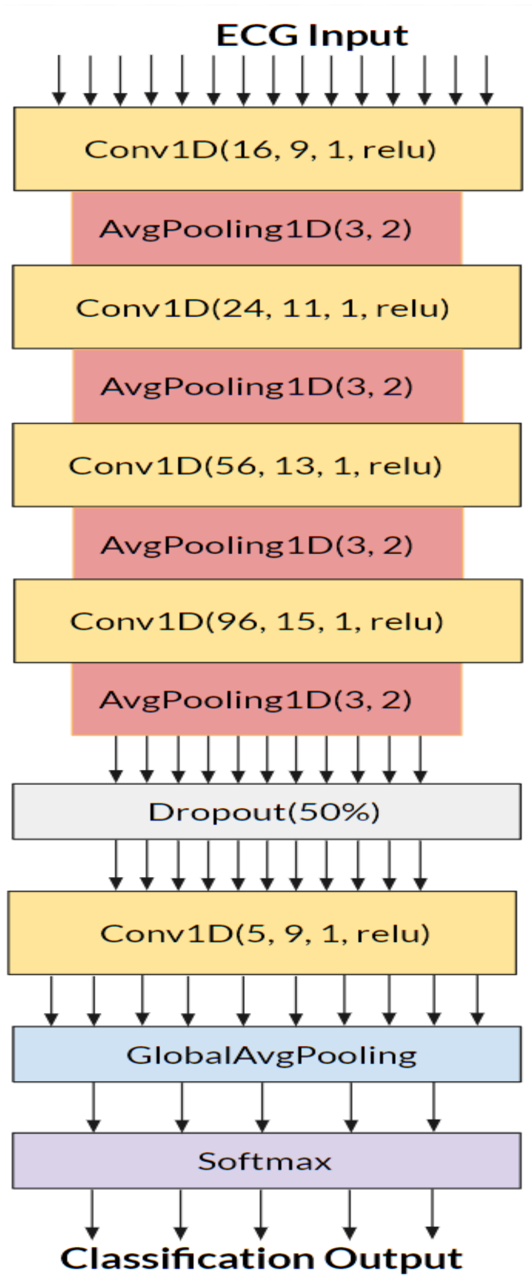

Figure 3.2: Modified 12-layer deep 1D CNN model

3.3 Proposed Hypothesis

Pruning of hyperparameters (kernel and filter size) was utilised to reduce the number of trainable/learnable parameters, and Global Average Pooling 1D was used in place of dense or completely connected layers.

The act of pruning involves deleting weighty connections from a network. The over-parameterized network is de-parameterized in this process. It is done to quicken inference and reduce the size of model storage. Kernel size and filter size in the first four CNN layers are hyperparameters that are pruned in the model.

$$P = (K * I + 1) * F \tag{3.1}$$

where P = Number Of Parameters In A Layer, K = Kernel size, I = Input channels, F = Number of filters and 1 is bias

Global average pooling: Usually for classification, the final convolutional layer's feature maps are vectorized and fed into fully connected layers, which are then followed by a softmax logistic regression layer. But overfitting is more prevalent in dense layers. Dropout(0.5) is used as a regularizer which zeroes out half of the activations to the fully linked layers during training, in order to prevent overfitting. However, we employed global average pooling for this project. [18] In the final convolution layer, it is intended to generate a feature map for each associated classification task category. We average each feature map, and the resulting vector is sent straight into the softmax layer, rather than constructing fully connected layers on top of the feature maps. By requiring correspondences between feature maps and categories, global average pooling is more native to the convolution structure than completely connected layers, which is one advantage. As a result, the feature maps can be simply understood as different types of confidence maps. Another benefit of global average pooling is that overfitting is prevented at this layer because there are no parameters to optimise. Additionally, because global average pooling sums up the spatial data, it is more resistant to input spatial translations.

3.4 About Dataset

The study uses the MIT-BIH arrhythmia database to assess performance. Each of the 48 records in the database contains two-channel ECG signals for 30 minutes that were chosen from 24-hour recordings of 47 different people. The database has 1,09,446 different sample numbers. The subjects were a total of 25 men between the ages of 32 to 89 and a total of 22 women between the ages of 23 and 89. The records 201 and 202 originated from the same male subject. Two leads are present on every recording; the modified limb lead II and one of the modified leads V1, V2, V4 or V5. Continuous ECG signals are band pass-filtered at 0.1—100 Hz and then digitized at 360 Hz. Twenty-three of the recordings (numbered in the range of 100-124) are intended to serve as a representative sample of routine clinical recordings and 25 recordings (numbered in the range of 200-234) contain complex ventricular, junctional, and supraventricular arrhythmias. The database contains annotations for both timing information and beat class information verified by independent experts.

1. Number of Samples: 100012
2. Number of Categories: 5
3. Sampling Frequency: 360 Hz
4. Major Categories: ['N': 0, 'S': 1, 'V': 2, 'F': 3, 'Q': 4]

Table 3.1: The MIT-BIH database, which was based on the AAMI, has categories for heartbeats (Association for the Advancement of Medical Instrumentation).

Category	Class
N	• Normal beat (N) • Left and right bundle branch block beats (L,R) • Atrial escape beat (e) • Nodal (junctional) escape beat (j)
S	• Atrial premature beat (A) • Aberrated atrial premature beat (a) • Nodal (junctional) premature beat (J) • Supraventricular premature beat (S)
V	• Premature ventricular contraction (V) • Ventricular escape beat (E)
F	• Fusion of ventricular and normal beat (F)
Q	• Paced beat (/) • Fusion of paced and normal beat (f) • Unclassifiable beat (U)

From Table 3.1 the five micro classes that we will be using for this project are:

- normal (NOR), symbol (.)
- left bundle branch block (LBBB), symbol (L)
- right bundle branch block (RBBB), symbol (R)
- Atrial premature beats (AP), symbol (A)
- premature ventricular beats (PVC), symbol (V)

3.5 Pre-Processing

Baseline drift, electrode contact noise, polarisation noise, internal amplifier noise, noise from muscle activity, and motor artefacts are only a few examples of the undesired noise and artefact effects that ECG data naturally contains. Noise was generated by the motion of the electrodes. In the field of ECG denoising, the bandpass filters, low-pass filters, and wavelet transforms are frequently utilised. Therefore, we must remove baseline wander and above noise in order to prepare the ECG data for the feature extraction step. We suggest applying wavelet filtering to the ECG signal since it can be used to compute the R-peak positions without affecting the original signal's position or shape. Sym4 from the Symlet wavelet function family was chosen as the wavelet function for this project [19]. Because CNN has the capability to automatically extract features from inside the signal, which improves network generalisation and lowers signal distortion, we used basic filtering.

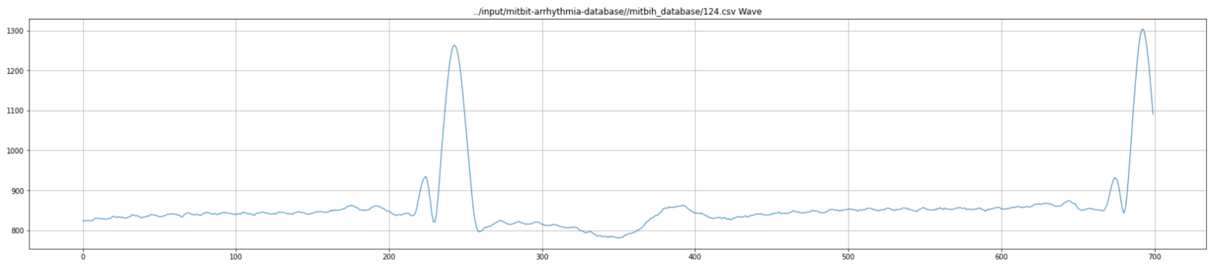

Figure 3.3: ECG signal before filtering

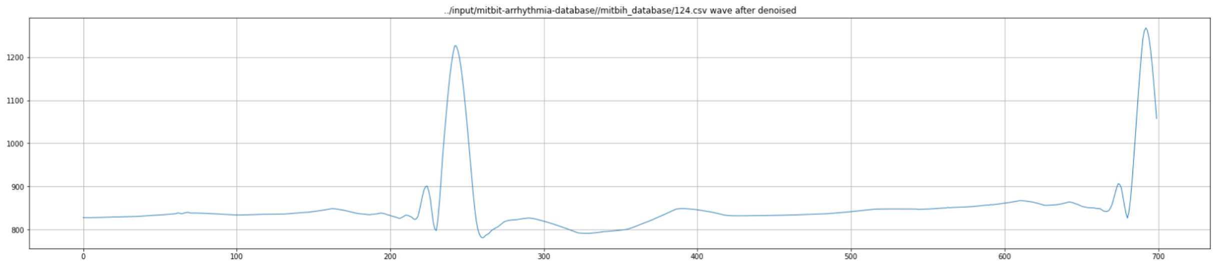

Figure 3.4: ECG signal after filtering

3.6 Data Segmentation

The MIT-BIH dataset annotates each heartbeat with a disease. This study divides heartbeats into five categories: normal, left and right bundle branch blocks, atrial premature beats, premature ventricular beats, and normal (PVC). To find R-peak at the start of the process, the Pan-Tompkins algorithm is utilised. The 360 samples that make up the dataset are focused on the identified R-peaks. The procedure selects a single lead from the dataset, and the Z-score method is used to normalise all segments.

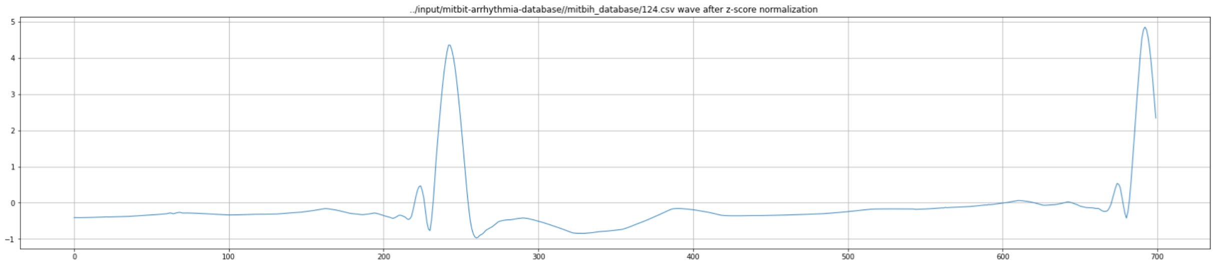

Figure 3.5: ECG signal after z-score normalization

3.7 Data Enhancement

An unbalanced dataset can result in overfitting for some of the classes, which has an impact on the model's training process. It also has an impact on CNN's network learning, which lowers recognition accuracy. In Table 3.2 after performing denoising and segmentation, we oversampled c2, c3, c4 and c5 to 15000 and undersampled c1 to 15000. Now all classes have equal weightage, hence risk of biasness towards predicting class with high weightage nullified.

Table 3.2: Unbalanced and balanced dataset.

Class	Type	Unbalanced	Balanced
C1	NOR	75011	15000
C2	LBBB	8071	15000
C3	RBBB	7255	15000
C4	AP	7129	15000
C5	PVC	2546	15000
		Total data = 100012	Total data = 75000

3.8 Understanding Parameters, Hyperparameters and GlobalAveragePooling

How is CNN trained? This brings up the issue of comprehending what a convolution neural network is used for, which is essentially an attempt to learn the values of filter(s) via backprop. In other words, a layer is "learnable" if it possesses weight matrices. In essence, the number of parameters in a particular layer is the count of "learnable" elements for a filter, or parameters for the filter for that layer, if such a word exists.
In general, parameters are weights that are learned during exercise. They are weight matrices that are modified during the back-propagation process and add to the predictive capacity of the model. Who controls the alteration? Well, they modify their values depending on the training technique you use, especially the optimization strategy.
Examples of parameters

1. the weights or coefficients of logistic and linear regression models.
2. neural network biases and weights
3. The centroids of the clustering.

$$P = (K * I + 1) * F \tag{3.2}$$

where P = Number Of Parameters In A Layer, K = Kernel size, I = Input channels, F = Number of filters and 1 is bias

Hyperparameters The learning algorithm uses hyperparameters when learning, but they are not included in the model that is produced. The trained model parameters, which are effectively what we refer to as the model, are what we have at the conclusion of the learning process. This model does not include the hyperparameters that were utilised during training. We only know the model parameters that were learned; for example, we cannot determine the hyperparameter values that were used to train a model from the model itself.

In machine learning and deep learning, a hyperparameter is essentially anything whose values or configuration are chosen before training begins and whose values or configuration will continue to be the same when training is complete.
Examples of Hyperparameters

1. Train-test split ratio
2. Learning rate in optimization algorithms (e.g. gradient descent)
3. Choice of optimization algorithm (e.g., gradient descent, stochastic gradient descent, or Adam optimizer)
4. The drop-out rate in nn (dropout probability)
5. Kernel or filter size in convolutional layers
6. Pooling size
7. Batch size

Global Average Pooling: [18]Instead of using CNN's standard fully connected layers for classification, we use a global average pooling layer to output the confidence of categories as the spatial average of the feature maps from the final mlpconv layer. The resulting vector is then fed into the softmax layer. Due to the fully connected layers in classic CNN, which function as a black box in between, it is challenging to understand how the category level information from the objective cost layer is transferred

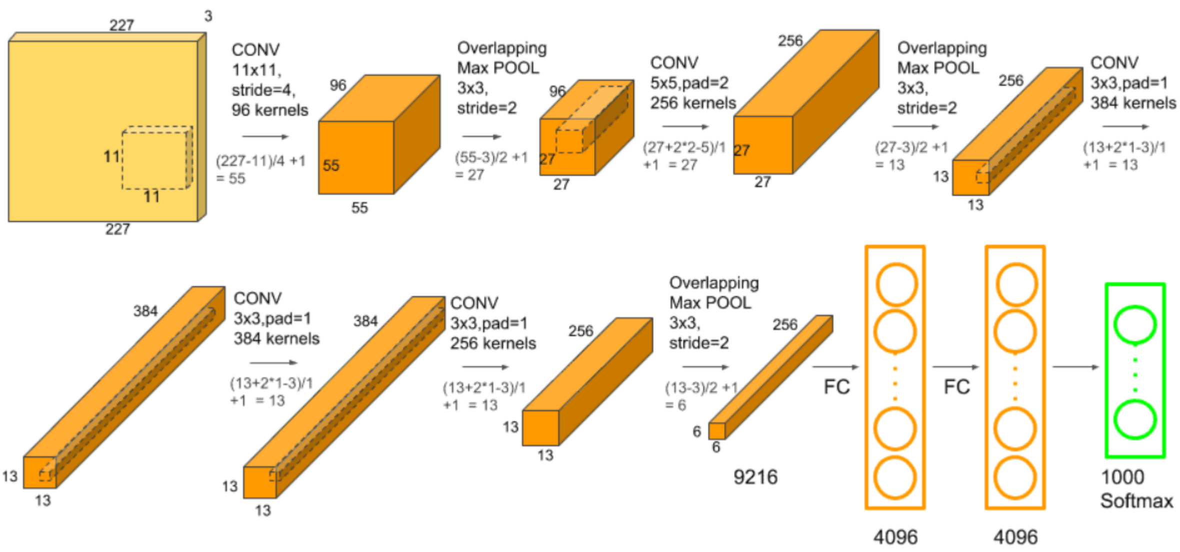

Figure 3.6: Pictorial representation of parameters and hyperparameters in a demo CNN model

back to the prior convolution layer. In contrast, global average pooling enforces correspondence between feature maps and categories, which is made feasible by a stronger local modelling utilising the micro network. As a result, global average pooling is more significant and understandable. Furthermore, the completely connected layers are vulnerable to overfitting and rely largely on dropout regularisation [4] [5], whereas global average pooling is a structural regularizer by itself and naturally guards against overfitting for the entire structure.

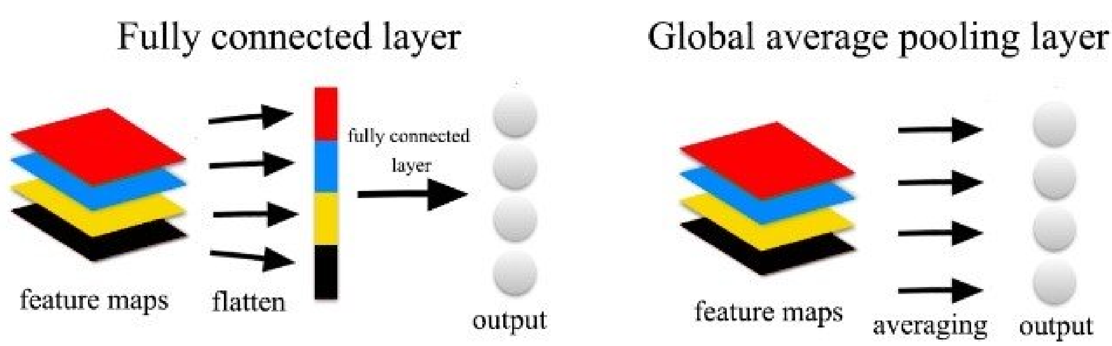

Figure 3.7: Dense layer vs Global average pooling layer

3.9 Mechanism

1. Input layer: At its essence, the input layer merely provides the shape of the input image; it has nothing to learn. So there are no learnable parameters here. As a result, there are no parameters.

2. CONV layer: We will undoubtedly have weight matrices because this is where CNN learns. Here, all we need to do is multiply the shape of width m by height n, account for all preceding layer's filters d, and then calculate the learnable parameters. For each of the filters, don't forget the bias word. A CONV layer would have the following number of parameters: ((m * n * d)+1)* k), with 1 added for each filter's bias term. The same expression can be expressed as ((shape of the width of the filter * shape of the height of the filter * number of filters in the previous layer+1)*number of filters). where "filter" refers to the quantity of filters in the active layer.

3. POOL layer: It only calculates a certain number; there is no backprop learning involved, hence there are no learnable parameters in this. As a result, there are no parameters.

4. Fully Connected Layer (FC):Undoubtedly, these characteristics can be learned. In fact, compared to the other levels, this category of layers has the most parameters. Since every neuron communicates with every other neuron, So how can we figure out how many parameters there are here? As you undoubtedly well know, it is the product of the number of neurons on the current layer c and the number of neurons on the preceding layer p. As always, do not forget the bias term. As a result, there are ((current layer neurons c * preceding layer neurons p)+1*c) parameters here.

5. Softmax layer: Like RELU, Tanh, or Sigmoid, Softmax is a parameter-free activation function that doesn't require training. Only the exponential of each logit is computed, and the resulting vector is then normalised by the exponential sum.

3.9.1 Implementation details

In this study, we suggested a modified 12-layer depp CNN in comparison to [6] in order to identify five micro classes included in the arrhythmia dataset while also considerably reducing the amount of trainable parameters in comparison to [6], from 292,887 to 106,997. Multiple changes were made in [6]. a network's structure:-

1. In the first layer(CNN layer), kernel size is changed from 13 to 9, rest keeping all things the same i.e filter size =16, strides = 1, activation function = rectified linear activation unit(relu), input shape = 360.

   (a) Parameters in 12-layer depp CNN are - $(13 \times 1 + 1) \times 16 = 224$
   (b) Parameters in modified 12-layer depp CNN are - $(9 \times 1 + 1) \times 16 = 160$

2. In the second layer(pooling layer), pool size and strides size remain same, i.e 3 and 2 respectively.
   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0
3. In the third layer(CNN layer), kernel size is reduced to 11 from 15 and filter size is reduced to 24 from 32, rest keeping all things the same i.e strides = 1, activation function = rectified linear activation unit(relu).
   (a) Parameters in 12-layer depp CNN are - $(15 \times 16 + 1) \times 32 = 7712$
   (b) Parameters in modified 12-layer depp CNN are - $(11 \times 16 + 1) \times 24 = 4248$
4. In the fourth layer(pooling layer), pool size and strides size remain same, i.e 3 and 2 respectively.
   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0
5. In the fifth layer(CNN layer), kernel size is reduced to 13 from 17 and filter size is reduced to 56 from 64, rest keeping all things the same i.e strides = 1, activation function = rectified linear activation unit(relu).
   (a) Parameters in 12-layer depp CNN are - $(17 \times 32 + 1) \times 64 = 34880$
   (b) Parameters in modified 12-layer depp CNN are - $(13 \times 24 + 1) \times 56 = 17528$
6. In the sixth layer(pooling layer), pool size and strides size remain same, i.e 3 and 2 respectively.
   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0
7. In the seventh layer(CNN layer), kernel size is reduced to 15 from 19 and filter size is reduced to 96 from 128, rest keeping all things the same i.e strides = 1, activation function = rectified linear activation unit(relu).
   (a) Parameters in 12-layer depp CNN are - $(19 \times 64 + 1) \times 128 = 1,55,776$
   (b) Parameters in modified 12-layer depp CNN are - $(15 \times 56 + 1) \times 96 = 80,736$
8. In the eighth layer(pooling layer), pool size and strides size remain same, i.e 3 and 2 respectively.

   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0

9. In the ninth layer, dropout is kept the same, i.e 0.5.
   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0

10. In the tenth layer, instead of using the first Dense layer, we used a convolution layer with 5 nodes as input(i.e filter size = 5), kerne size = 9 and activation function = rectified linear activation unit(relu).
   (a) Parameters in 12-layer depp CNN are - $(35 \times 2688 + 1 \times 35) = 94, 115$
   (b) Parameters in modified 12-layer depp CNN are - $(9 \times 96 + 1) \times 5 = 4325$

11. In the eleventh layer, instead of using the second Dense layer, we used a GlobalAveragePooling1D. So instead of using dense layer we used global average pooling.
   (a) Parameters in 12-layer depp CNN are - $(5 \times 35 + 1 \times 5) = 180$
   (b) Parameters in modified 12-layer depp CNN are - 0

12. In the twelfth layer(softmax layer).
   (a) Parameters in 12-layer depp CNN are - 0
   (b) Parameters in modified 12-layer depp CNN are - 0

CHAPTER 4

Experiments and results

This section contains all the experiments performed and generated by the proposed model.

4.1 Metrics used

1. Accuracy

   The base metric used for model evaluation is often Accuracy, describing the number of correct predictions over all predictions. Accuracy is calculated as :

$$Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + FalsePositive + FalseNegative + TrueNegative} \tag{4.1}$$

2. Precision

   Precision give the score that how much times the class is predicted correctly to the number of samples of the class. Precision is calculated as :

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{4.2}$$

3. Recall

   Recall give the score that how much times the class is predicted correctly to the total number of times the class is predicted. Recall is calculated as :

$$Recall = \frac{TruePositive}{TruePositive + FalseNegatives} \tag{4.3}$$

4. F1 score

   F1 score is generally the average of Recall and precision but it is weighted. F1 score is calculated as :

$$F1score = \frac{2 \times (precision \times recall)}{precision + recall} \tag{4.4}$$

4.2 Experiment design

4.2.1 Modified 12-layer deep 1D convolutional neural network

4.2.1.1 Experiment description

In total the model comprises of total 12 layers. Starting 8 layers are convolutional and pooling layer arranged alternately, so total 4 convolutional and 4 pooling layers are used alternately. Then a dropout layer is used, after that two dense layers and at last softmax activation function layer. The model ran a total of 20 epochs with a batch size of 36.

Each convolutional layer has filter, kernel, strides, padding and activation function. Each pooling layer has pool size and strides size. Layer 1(cnn layer) has 16 filters, kernel size 9, padding = 'same'(means for a convolution with a stride=1, (or for pooling) it should produce output of the same size as the input), activation function = 'relu'. Layer 2(pooling layer) has pool size = 3, strides = 2. Layer 3(cnn layer) has 24 filters, kernel size 13, padding ='same', activation function = 'relu'. Layer 4(pooling layer) has pool size = 3, stides = 2. Layer 5(cnn layer) has 56 filters, kernel size 15, padding ='same', activation function = 'relu'. Layer 6(pooling layer) has pool size = 3, stides = 2. Layer 7(cnn layer) has 96 filters, kernel size 17, padding ='same', activation function = 'relu'. Layer 8(pooling layer) has pool size = 3, stides = 2. Layer 9(droput layer) has dropout of 50% to set off half of the activation nodes, to avoid overfitting. Layer 10(cnn layer) number of output classes=5, kernel size= 9, strides = 1, activation function = 'relu' . Layer 11(global average pooling layer) which is used as alternative to dense layers, since global average pooling averages out the feature map, hence problem for overfitting is also reduced. Layer 12(softmax layer) has 5 inputs and it will predict 5 outputs, it is used to compute the exponents of every logit and then normalize the output vector by the sum of exponents.

4.2.1.2 Results and Discussion

From table 4.1 and 4.2, trainable parameters are reduced from 292,887 to 106,997 in modified 12-layer deep one dimensional model.

Table 4.1: Summary table of 12-layer deep one dimensional model

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_5 (Conv1D)            (None, 360, 16)           224
_________________________________________________________________
average_pooling1d_4 (Average (None, 179, 16)           0
_________________________________________________________________
conv1d_6 (Conv1D)            (None, 179, 32)           7712
_________________________________________________________________
average_pooling1d_5 (Average (None, 89, 32)            0
_________________________________________________________________
conv1d_7 (Conv1D)            (None, 89, 64)            34880
_________________________________________________________________
average_pooling1d_6 (Average (None, 44, 64)            0
_________________________________________________________________
conv1d_8 (Conv1D)            (None, 44, 128)           155776
_________________________________________________________________
average_pooling1d_7 (Average (None, 21, 128)           0
_________________________________________________________________
flatten (Flatten)            (None, 2688)              0
_________________________________________________________________
dropout_1 (Dropout)          (None, 2688)              0
_________________________________________________________________
dense (Dense)                (None, 35)                94115
_________________________________________________________________
dense_1 (Dense)              (None, 5)                 180
_________________________________________________________________
softmax_1 (Softmax)          (None, 5)                 0
=================================================================
Total params: 292,887
Trainable params: 292,887
Non-trainable params: 0
_________________________________________________________________
```

Table 4.2: Summary table of modified 12-layer deep one dimensional model

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 360, 16)           160
_________________________________________________________________
average_pooling1d (AveragePo (None, 179, 16)           0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 179, 24)           4248
_________________________________________________________________
average_pooling1d_1 (Average (None, 89, 24)            0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 89, 56)            17528
_________________________________________________________________
average_pooling1d_2 (Average (None, 44, 56)            0
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 44, 96)            80736
_________________________________________________________________
average_pooling1d_3 (Average (None, 21, 96)            0
_________________________________________________________________
dropout (Dropout)            (None, 21, 96)            0
_________________________________________________________________
conv1d_4 (Conv1D)            (None, 21, 5)             4325
_________________________________________________________________
global_average_pooling1d (Gl (None, 5)                 0
_________________________________________________________________
softmax (Softmax)            (None, 5)                 0
=================================================================
Total params: 106,997
Trainable params: 106,997
Non-trainable params: 0
_________________________________________________________________
```

4.2.1.3 Classification Report

A classification report is generated after the model is trained completely. Class-wise metrics are given including the precision, F1 score and recall. Support is the total number of sample of the class in training set. Table 4.3 and table 4.4 represents the classification report of previous and proposed model respectively. In table 4.3, test accuracy found to be 98.16%. In table 4.4 test accuracy found to be 96.35%.

Table 4.3: Classification report of 12-layer deep one-dimensional convolutional neural network

micro classes	Precision	Recall	F1-score	Support
N	0.96	0.98	0.96	2997
L	0.98	0.96	0.96	2995
R	0.96	0.96	0.96	3083
A	0.95	0.96	0.95	2996
V	0.96	0.98	0.96	2929
Test accuracy			98.16%	15000

Table 4.4: Classification report of modified 12-layer deep one-dimensional convolutional neural network

micro classes	Precision	Recall	F1-score	Support
N	0.92	0.94	0.92	2997
L	0.94	0.92	0.92	2995
R	0.94	0.96	0.94	3083
A	0.97	0.96	0.96	2996
V	0.93	0.94	0.93	2929
Test accuracy			96.35%	15000

4.2.1.4 Graphical Comparisons

Figure 4.1, 4.3 illustrates the comparison of loss and epoch for the 12-layer deep 1-D CNN model and the modified 12-layer deep 1-D CNN model, respectively, on both the training and test data. A loss is a numerical representation of how poorly the model predicted an example. The loss is zero if the model's forecast is accurate; otherwise, the loss is higher. A single epoch is the single forward and reverse pass through the neural network of the complete dataset. We split up an epoch into several smaller batches since it would be too large to feed the machine all at once.

Figure 4.2, 4.4 displays the comparison of accuracy and epoch for the 12-layer deep 1-D CNN model and the modified 12-layer deep 1-D CNN model, respectively, on both the training and test data. By calculating the percentage difference between the model predictions and the actual values, accuracy assesses how effectively our model predicts the future.

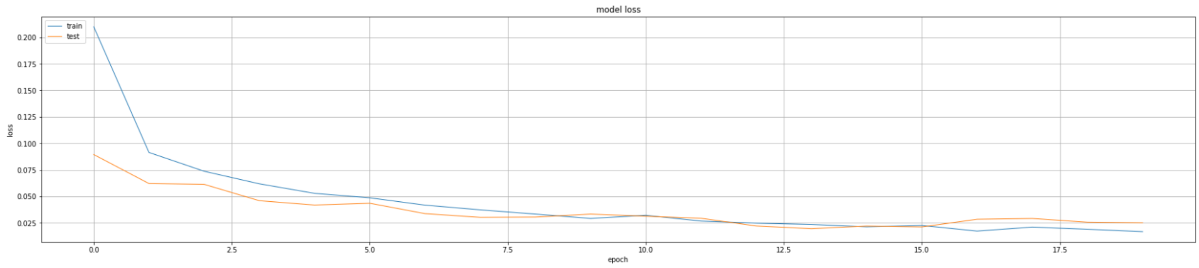

Figure 4.1: Loss v/s Epoch graph of 12-layer deep 1D CNN model to get the overall idea of loss occured for 20 epochs.

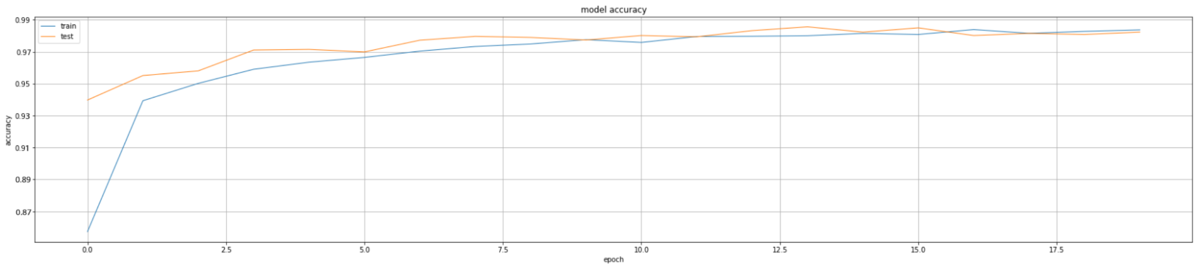

Figure 4.2: Accuracy v/s Epoch graph of 12-layer deep 1D CNN model to get the overall idea of accuracy for 20 epochs.

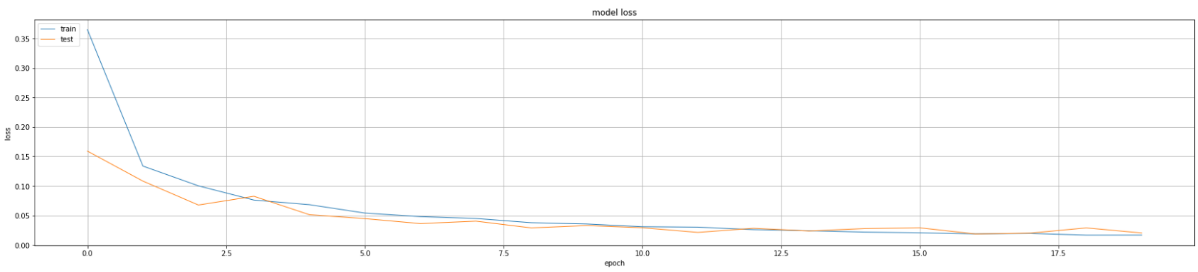

Figure 4.3: Loss v/s Epoch graph of modified 12-layer deep 1D CNN model to get the overall idea of loss occured for 20 epochs.

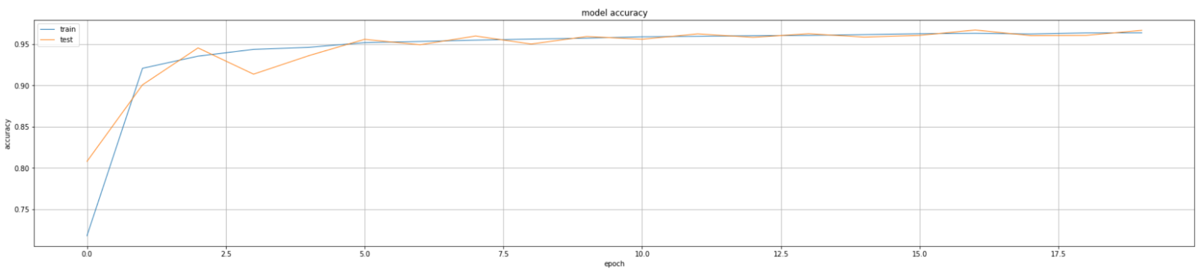

Figure 4.4: Accuracy v/s Epoch graph of modified 12-layer deep 1D CNN model to get the overall idea of accuracy for 20 epochs.

CHAPTER 5

Conclusion

This section specifies the contributions,limitations and future scope of this project.

5.1 Contributions

In the modern world, cardiovascular disease is a serious health issue. The ECG is crucial for making an early diagnosis of cardiac arrhythmia. Since skilled medical resources are scarce, it can be difficult and time-consuming to visually recognise the ECG signal. Unlike the current research, which was primarily concerned with enhancing accuracy regardless of extremely high computing cost. Very sophisticated equipment are needed to perform enormous computations, but these machines are very expensive and not scalable to be readily available at every medical facility centre. So, in this project, we work to create a model that attempts to maximise accuracy while minimising computational cost. Pruning, which entails reducing the filters and kernel size of convolutional layers and employing one-dimensional global average pooling rather than fully connected layers, is how we reduced model complexity. This naturally lowers the overfitting issue. Although it can't completely replace expert analysis owing to some accuracy loss, by putting on smart watches or mobile devices, it can at least provide earlier emergency assistance. Also, it would enable future consultations to be conducted by knowledgeable ECG signal analysts. Additionally, in the MIT-BIH Arrhythmia database, we paid more attention to classifying specific micro-classes, such as the Normal, Left Bundle Branch Block, Right Bundle Branch Block, Atrial Premature Beats, and Premature Ventricular Beats, rather than the five main classes Non-ectopic, Supraventricular ectopic, Ventricular ectopic, Fusion, and Unknown.

5.2 Limitations

More reduction in filters and kernel size will decrease the trainable parameters, but there may be the chances of model behaving abruptly, i.e accuracy may see a significant drop. So reduction in trainable parameters must be done cautiously.

5.3 Future scope

Future models may be made even simpler using additional techniques, such as compressing the model using the concept of knowledge distillation. Knowledge distillation is the process of moving information from a large, cumbersome model or set of models to a single, more manageable model that may be used in real-world applications. The concept of quantization is also applicable. Weights are kept as 32-bit floating-point values in deep neural networks. Quantization is the concept of using fewer bits to encode certain weights. Weights may be quantized using 16-bit, 8-bit, 4-bit, or even 1-bit technology. The size of the deep neural network can be greatly decreased by lowering the amount of bits utilised.

REFERENCES

[1] K. Mc Namara, H. Alzubaidi, and J. K. Jackson, "Cardiovascular disease as a leading cause of death: how are pharmacists getting involved?" *Integrated pharmacy research & practice*, vol. 8, p. 1, 2019.

[2] W. Caesarendra, T. A. Hishamuddin, D. T. C. Lai, A. Husaini, L. Nurhasanah, A. Glowacz, and G. A. F. Alfarisy, "An embedded system using convolutional neural network model for online and real-time ecg signal classification and prediction," *Diagnostics*, vol. 12, no. 4, p. 795, 2022.

[3] S.-L. Guo, L.-N. Han, H.-W. Liu, Q.-J. Si, D.-F. Kong, and F.-S. Guo, "The future of remote ecg monitoring systems," *Journal of geriatric cardiology: JGC*, vol. 13, no. 6, p. 528, 2016.

[4] W. Yin, X. Yang, L. Zhang, and E. Oki, "Ecg monitoring system integrated with ir-uwb radar based on cnn," *IEEE Access*, vol. 4, pp. 6344–6351, 2016.

[5] S.-N. Yu and Y.-H. Chen, "Electrocardiogram beat classification based on wavelet transformation and probabilistic neural network," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1142–1150, 2007.

[6] M. Wu, Y. Lu, W. Yang, and S. Y. Wong, "A study on arrhythmia via ecg signal classification using the convolutional neural network," *Frontiers in Computational Neuroscience*, vol. 14, 2021, doi: 10.3389/fncom.2020.564015. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fncom.2020.564015

[7] S. Y. Wong, K. S. Yap, H. J. Yap, and S. C. Tan, "A truly online learning algorithm using hybrid fuzzy artmap and online extreme learning machine for pattern classification," *Neural Processing Letters*, vol. 42, no. 3, pp. 585–602, 2015.

[8] S. Y. Wong, K. S. Yap, H. J. Yap, S. C. Tan, and S. W. Chang, "On equivalence of fis and elm for interpretable rule-based knowledge representation," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 7, pp. 1417–1430, 2014.

[9] S. Y. Wong, K. S. Yap, and H. J. Yap, "A constrained optimization based extreme learning machine for noisy data regression," *Neurocomputing*, vol. 171, pp. 1431–1443, 2016.

[10] P. Pławiak and U. R. Acharya, "Novel deep genetic ensemble of classifiers for arrhythmia detection using ecg signals," *Neural Computing and Applications*, vol. 32, no. 15, pp. 11 137–11 161, 2020.

[11] M. Zubair, J. Kim, and C. Yoon, "An automated ecg beat classification system using convolutional neural networks," in *2016 6th international conference on IT convergence and security (ICITCS)*. IEEE, 2016, pp. 1–5.

[12] 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," *Information Sciences*, vol. 415, pp. 190–198, 2017.

[13] D. K. Atal and M. Singh, "Arrhythmia classification with ecg signals based on the optimization-enabled deep convolutional neural network," *Computer Methods and Programs in Biomedicine*, vol. 196, p. 105607, 2020.

[14] X. Xu and H. Liu, "Ecg heartbeat classification using convolutional neural networks," *IEEE Access*, vol. PP, pp. 1–1, 01 2020, doi: 10.1109/ACCESS.2020.2964749

[15] T. Wang, C. Lu, Y. Sun, M. Yang, C. Liu, and C. Ou, "Automatic ecg classification using continuous wavelet transform and convolutional neural network," *Entropy*, vol. 23, p. 119, 01 2021, doi: 10.3390/e23010119

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, p. 2012.

[17] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR*, vol. abs/1207.0580, 2012, cite arxiv:1207.0580. [Online]. Available: http://arxiv.org/abs/1207.0580

[18] M. Lin, Q. Chen, and S. Yan, "Network in network," 12 2013.

[19] B. N. Singh and A. K. Tiwari, "Optimal selection of wavelet basis function applied to ecg signal denoising," *Digital signal processing*, vol. 16, no. 3, pp. 275–287, 2006.

ORIGINALITY REPORT

14%	10%	8%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.slideshare.net Internet Source	3%
2	www.ncbi.nlm.nih.gov Internet Source	2%
3	Submitted to ABV-Indian Institute of Information Technology and Management Gwalior Student Paper	1%
4	Submitted to Liverpool John Moores University Student Paper	1%
5	www.coursehero.com Internet Source	1%
6	Submitted to The Robert Gordon University Student Paper	1%
7	towardsdatascience.com Internet Source	1%
8	Submitted to National University of Singapore Student Paper	1%