REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

DEEP LEARNING FOR ECG SIGNAL CLASSIFICATION IN REMOTE HEALTHCARE APPLICATIONS

Sura Ali Al-TIMIMI

Master's Thesis

Supervisor

Prof. Dr. Hasan Hüseyin BALIK

İstanbul, 2023

DEEP LEARNING FOR ECG SIGNAL CLASSIFICATION IN REMOTE HEALTHCARE APPLICATIONS

Sura Ali AL-TIMIMI

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2023

The thesis titled DEEP LEARNING FOR ECG SIGNAL CLASSIFICATION IN REMOTE HEALTHCARE APPLICATION prepared by SURA ALI HASHIM AL-TIMIMI and submitted on 23/08/2023 has been **accepted unanimously** the degree of Master of Science in Electrical and Computer Engineering.

Prof. Dr Hasan Hüseyin BALIK

Supervisor

Thesis Defense Jury Members:

Prof. Dr. Hasan Hüseyin BALIK	Department of Computer Engineering, Istanbul Aydın University	________________
Assoc. Prof. Dr. Oğuz ATA	Department of Software Engineering, Altinbaş University	________________
Asst. Prof. Dr. Aytuğ BOYACI	Department of Computer Engineering, National Defence University	________________

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis.

Submission date of the thesis to Institute of Graduate Studies: ___/___/____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Sura Ali AL-TIMIMI

Signature

DEDICATION

I dedicate this research work and pledge it to my prominent supervisor in guiding me throughout the entire research work, as well as the person who stood with me during my academic career and during everything in my life, whom I consider more than a father to me, Dr. Mustafa Al-Hassow .

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisor Prof. Dr. Hasan BALIK for guiding and helping me along the way in writing this dissertation. Discussing my progress, problems, and ideas with my supervisor Prof. Dr. Hasan BALIK a couple of times every week helped me tremendously in understanding the logic behind the research. It made me better realize the technical need for this research work.

ABSTRACT

DEEP LEARNING FOR ECG SIGNAL CLASSIFICATION IN REMOTE HEALTHCARE APPLICATIONS

HASAN, Sura Ali

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor. Prof. Dr. Hasan Hüseyin BALIK

Date: August / 2023

Pages: 54

The thesis looks at how to improve the work of ECG devices using deep learning algorithms. This project's goal is to help doctors detect heart disorders and diseases that occur in the heart faster that can cause a person's death. The one-dimensional convolutional neural network algorithm was used, which is considered one of the best algorithms for this work. The model was applied to the MIT-BIH database. assigned to this work. This work was compared with another work in which a different algorithm was used, and it reached a lower accuracy rate than our current work. In this work, we reached an accuracy of 98%, which is considered one of the best results that we reached. Also, the loss rate was 0.2%. The model was applied in the Python language, which is considered one of the best languages for this work, as the Python language is considered one of the most flexible and smooth languages, especially when the algorithm is related to deep learning.

Keywords: ECG, Deep Learning, CNN, Classification, Heart Arrhythmias.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

AL	:	Artificial Intelligence
ECG	:	Electrocardiogram
DL	:	Deep Learning
RELU	:	Rectified Linear Activation Function
1D	:	One Dimensional
SVM	:	Support Vector Machine
CNN	:	Convolutional Neural Network
REGPAR	:	Receptive Field Paring
RNN	:	Recurrent Neural Network
LSTM	:	Long Short-Term Memory
MLPNN	:	Multi-Layer Perceptron Neural Network
ANN	:	Artificial Neural Network
KNN	:	K-Nearest Neighbors Algorithm
PCA	:	Principal Component Analysis
NCA	:	Neighborhood-Component-Analysis

1. INTRODUCTION

1.1 ELECTROCARDIOGRAM (ECG)

In the beginning, we will talk about the ECG device in general and how this device works. The main cause of mortality worldwide is heart disease. According to estimates, in 2017, 17.8 million people died from heart disease globally, making up close to 31% of all fatalities. World Health Organization data on human mortality. For effective therapy to work and to lower mortality, early identification of heart disease is essential. Electrocardiography is a low-cost, quick method that helps us understand how the heart works and, in turn, aids in the diagnosis of cardiac disorders. A cardiologist utilizes an electrocardiogram (ECG), which records the electrical activity of the heart, to discover abnormal cardiac function and diagnose a number of ailments. Unfortunately, analyzing an ECG recording takes a lot of time and the concentration and focus of an experienced professional. The ECG device is a magnetic electrode that is placed directly on the skin. These electrodes, in turn, detect any disturbance in the heart and produce it on a chart. He took his idea of the galvanometer responsible for the intensity of the heart. electric current, but with a significant development of the device over the passage of time. Disorders that can affect the heart, such as atrial fibrillation and ventricular arrhythmias, as well as insufficient coronary blood flow due to myocardial ischemia and myocardial infarction. What the device measures is the connection of neurons with each other and is shown in the form of a graphic. This diagram is what is called an electrocardiogram. This is done by electrical and chemical signals [31].

A standard ECG contains ten lead wires that give twelve views of the heart. An ECG can measure your heart rate, rhythm, force, and timing of electrical signals. The test can be used to evaluate many conditions related to the heart. When the heart muscle cells are at rest, they are considered depolarized because the concentration of ions across the cell membranes is uneven and no electrical activity occurs. Ions like sodium (Na^+), potassium (K^+), and calcium (Ca^{2+}) are at resting potential and have varying concentrations both within and outside the cell. When there is an electrical impulse, these ions cross the cell membrane and cause a depolarization or action potential. Depolarization causes the heart to contract. Repolarization occurs when these ions return to their resting state. Depolarization and repolarization of the heart maintain regular contractions of the heart and blood circulation throughout the body. The heart has two atria and two ventricles, making up its four chambers.

The ECG consists of waves that represent the sequence of depolarization and repolarization of the atria and ventricles. P Wave: Atrial Depolarization Associated Small Deflection Wave Between the initial deflection of the P wave and the first deflection of the QRS complex is the PR interval. Three waves that correlate to ventricular depolarization make up the QRS complex. The ventricular repolarization waveform is the T-wave. Atrial repolarization is prevented by the QRS complex. QT interval- the time between the commencement of the QRS wave to the end of the T wave. The cycle from ventricular depolarization to ventricular repolarization is represented by it [1].

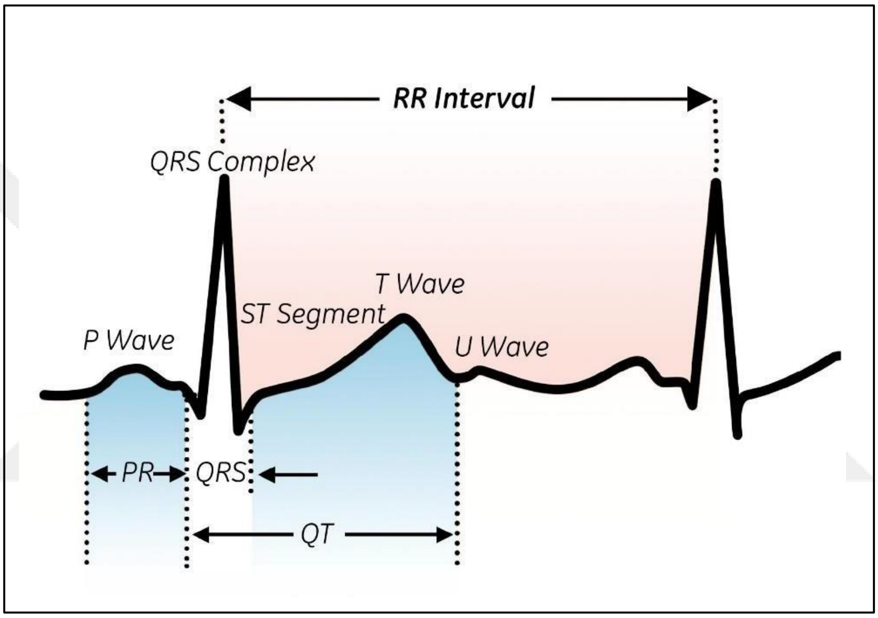

Figure 1.1: ECG Waves.

Convolutional neural networks and deep learning have played a significant role in the development of ECGs. Deep learning contains enough algorithms and intelligence to greatly improve the work of these devices. It has achieved great success in recognizing and diagnosing many diseases through its ability to extract traits without prior treatment. And through the work of the neural network with high accuracy and professionalism, with its layers, and with progress in the availability of digital data and computing power. Provides deep learning in addition to improving and enhancing ECG accuracy, it also extends its core functionality.

1.2 ARTIFICIAL INTELLIGENCE

The study and creation of intelligent devices, typically computer programs, with the aim of imitating human intelligence is known as artificial intelligence (AI). In other words, it helps with decision-making and problem-solving. A branch of study that leverages both computer science and large datasets. Artificial intelligence may operate in the digital world thanks to the accessibility of digital devices and specialized software for algorithm analysis and creation, machine learning, and generally, the artificial intelligence system absorbs vast volumes of training data. A task that might benefit from training data is the identification and description of objects in photos by evaluating millions of instances saved by a smart device. A school of thought known as communicative started studying artificial intelligence in 1940 AD, which led to the study of the thought process. In 1950, Alan Turing presented a research paper in which he studied a thinking machine that imitates a person without having any obviously different features in it. Later, in the year 1950 A.D., Hodgkin Huxley presented a model that uses an electrical network to symbolize neurons and an electric current to represent impulses that activate or deactivate cells to represent the human brain. Artificial intelligence (AI) was birthed at a Dartmouth College meeting in the year 1956 AD thanks to the work of these researchers and their models. Research into artificial intelligence was placed on hold for a while due to a lack of high speeds and storage capacities but picked back up in the eighties when the United States and Britain introduced the fifth-generation initiative in computer technology. The smart agent, which is used in things like news retrieval services, online shopping, and web browsing, was developed as part of AI research in the early 1990s. Today, AI is being used in previously unimaginable contexts, such as providing physical assistance in the form of bots, being embedded in customer service software, answering the phone, and more [32].

1.3 MACHINE LEARNING

Now we'll talk about one subfield of AI: machine learning. Data (training data) and programs play a central role in this field of study, which attempts to simulate the way humans learn and develop their abilities over time. Machine learning, which employs statistical models and algorithms for learning, seeks to give computers cognitive abilities that are competitive with those of humans. The point of machine learning Thus, machine learning enables computer programs to access and use data directly, without human intervention in the process, by making systems capable of learning and developing automatically, through experience, without the need to conduct software operations. There are three primary components to machine learning: the underlying computational algorithm, the data, and the output. The factors and considerations that go into choosing a choice. Source of information from which the solution is drawn. Machine learning is now a component of every web application because businesses employ it to address a wide range of issues. This is because judgments made based on data would determine whether or not the program would keep up with progress or lag behind it. Data is the true basis for any work that can be done in life. Moreover, as technologies in this area progress swiftly and the scope of machine learning expands to include many capabilities, its practical applications can quickly increase business profits. Industries, for instance, typically contain vast amounts of data and information, necessitating specialized systems for efficient and accurate analysis; these sectors have generally come to accept the machine as the most effective tool for constructing models, formulating strategies, and organizing for the future. Signal recognition is one of the most well-known uses of machine learning that has become essential to modern human living. It may be used to recognize individuals, digital photos, places, and other items. One of the most well-known examples of contemporary machine learning in action is Google's voice search, which uses speech recognition to translate spoken commands into text. [33].

1.4 DEEP LEARNING

Essentially, a neural network with three or more tiers constitutes the deep learning subfield within machine learning. Although these neural networks can't learn as efficiently from large datasets as the human brain can, they try to mimic its behavior. A neural network with only one layer can still generate approximations, but adding hidden layers improves accuracy. Deep learning is used by many AI apps and services to enhance automation by taking over analytical and physical activities that previously needed human input. Both established and forthcoming technologies rely on deep learning techniques, including digital companions, voice-enabled TV remote controls, and credit card fraud detection (such as self-driving cars). The data it uses and the methods it employs to learn set deep learning apart from traditional machine learning. To make predictions, machine learning algorithms pick features from the input data model and organize them into tables based on labels. Even so, this does not prove that it does not employ unorganized information. Pre-processing is commonly used to organize the data into a more conventional framework. Some of the data pre-processing steps typically used in machine learning can be skipped when using deep learning. In this way, these algorithms can automate feature extraction from unstructured data like text and signals, reducing the need for human specialists. Let's say, for the sake of argument, that we have a number of pictures of various animals and we'd like to sort them into cat, canine, hamster, etc. categories. In order to tell one species from another, deep learning algorithms can figure out which characteristics, like ears, are the most telling. This feature hierarchy is typically developed by hand in machine learning, with the help of a human specialist. The deep learning algorithm then fine-tunes and "fits" itself through gradient regression and backpropagation, improving the precision with which it predicts the appearance of the animal in subsequent signals. Both supervised and autonomous learning, as well as reinforcement learning, are within the capabilities of ML and DL models. In order to properly classify or predict data, supervised learning relies on labeled datasets, which necessitates human intervention. Unsupervised learning, on the other hand, can work with raw data without any labels at all by looking for trends and then classifying the data accordingly. A deep neural network receives its input and output from the visual levels. Using data from the input layer, a deep learning model creates a final forecast or classification at the output layer. Another technique for training a model is called backpropagation; it uses techniques like gradient descent to determine prediction errors and then modifies the weights and biases of the function by rotating the training process across the layers. The neural network's ability to anticipate and adjust for mistakes is due to the collaborative efforts of forward propagation and backpropagation. The program improves its precision over time. All of the above is a simplified description of the most basic forms of deep neural networks. But deep learning algorithms are extremely intricate, and there are numerous varieties of neural networks tailored to particular tasks and data sets. Commonly used in computer vision applications and picture classification, CNNs are capable to identifying objects and other details in pictures. In 2015, CNN was the first to win an object recognition competition against a human competitor. Due to their ability to process sequential or time series data, recurrent neural networks (RNNs) find widespread use in speech identification and natural language processing. [34]

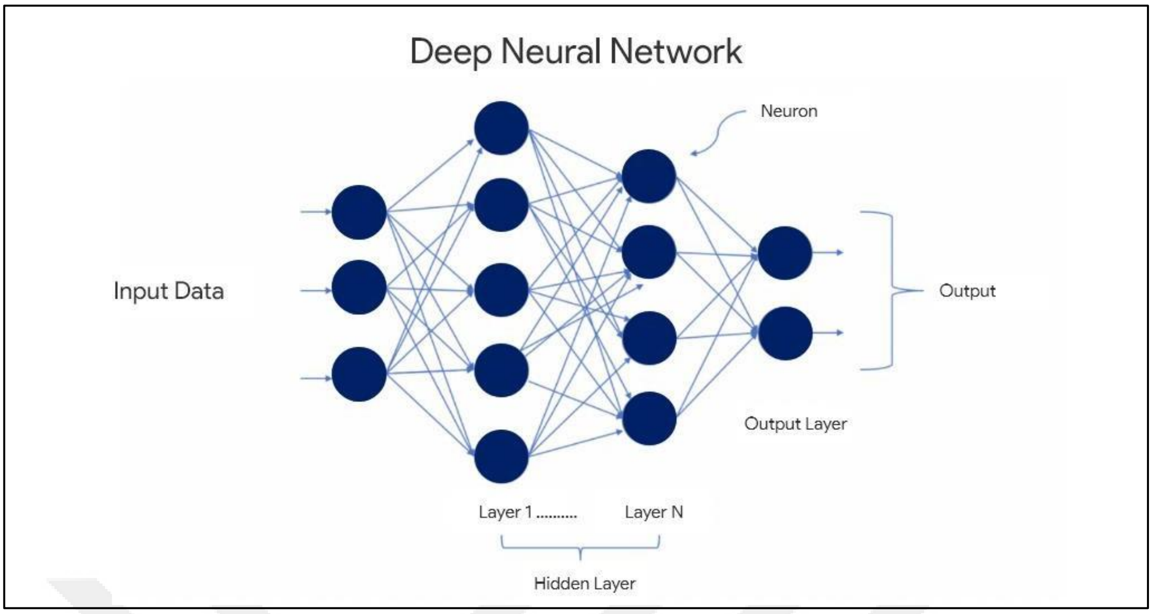

Figure 1.2: Deep Neural Network.

1.5 NEURAL NETWORKS (NN)

It is a machine learning network consisting of several levels. Each level represents a node, and each of these nodes includes one input layer, one or more hidden layers, and one other layer. Each node has a weight and a threshold connected to each other. Every output that appears from the node and this output exceeds the threshold value, the data begins to be submitted to the upper layer of this network, and if this does not happen, the process of sending any data to the next network (the second level) fails. The neural network contains several types, including the convolutional neural network (CNN) and the recurrent neural network (RNN). Each of these types is used for different data. The convolutional neural network (CNN) is used for classification and analysis data such as images and digital data, while the recurrent neural network (RNN) is used for natural language, speech recognition, and video. And others. In our work, and for the type of data we deal with, a convolutional neural network is the best choice. Before the invention of the neural network, the process of extracting data was very difficult as this process was done manually, but after the invention of this network, this data was extracted using matrix multiplication and other concepts from linear algebra, this greatly facilitated this process. There are three primary sorts of layers in them:

a. Convolutional layer.

b. Pooling layer.

c. Fully connected (FC) layer.

1.6 CONVOLUTIONAL NEURAL NETWORK (CNN)

It is a mathematical algorithmic model that mimics the functional characteristics of organic brain networks. It has neurons, nodes, and synaptic connections between the nodes. A neural network consists of many nodes (or neurons or units) that are interconnected. Each neuron takes in a linear collection of information, executes a nonlinear switch (sometimes referred to as an activation function), and then sends it out as an output. Every two-node connection represents a weighted value called weight. Various weights and activation functions result in various neural network outputs. In the signal above, the middle layer is referred to as the hidden layer, the middle neuron is referred to as the output layer, and the leftmost original input information is referred to as the input layer. A lot of nonlinear input data is taken in by the neuron input layer. The input vector refers to the input data. Information is delivered, evaluated, and weighted in neuron connections at the output layer to provide output outcomes. The output vector refers to the output message. The hidden layer is a layer that connects the input layer and output layer and is made up of several neurons. Many buried layers indicate numerous activation mechanisms.

signal-related machine learning issues, particularly those involving large signals, are well-suited to the convolutional neural network. The convolutional network is possible to be trained because, through a sequence of procedures, the massive data volume is transformed into manageable dimensions of the signal recognition problem. Yann LeCun is widely credited with being the inventor of CNN and its first application, handwritten font detection (MINST). LeNet is the name LeCun has given his planned network. [35]

1.6.1 Convolutional Layer

Convolutional neural networks have a first layer, called the convolutional layer, whose main function is to search for features in the incoming input. This layer is completely ignored by the data. Just "drag" a window representing the feature onto the signal to apply the filter, and then figure out the convolution product between the window and each scanned region. In this way, the convolution filter achieves its effects. It's commonly understood that a feature is the same thing as a filter. The convolutional layer receives many input pictures and applies the filters to each one individually via convolution. Our picture filters are tailored to the specific characteristics we seek. They help us achieve enhanced outcomes for the input signals or data. For each signal filter set, we obtain a feature map indicating feature locations; a greater value indicates a closer match between the feature and the associated pixel in the picture.

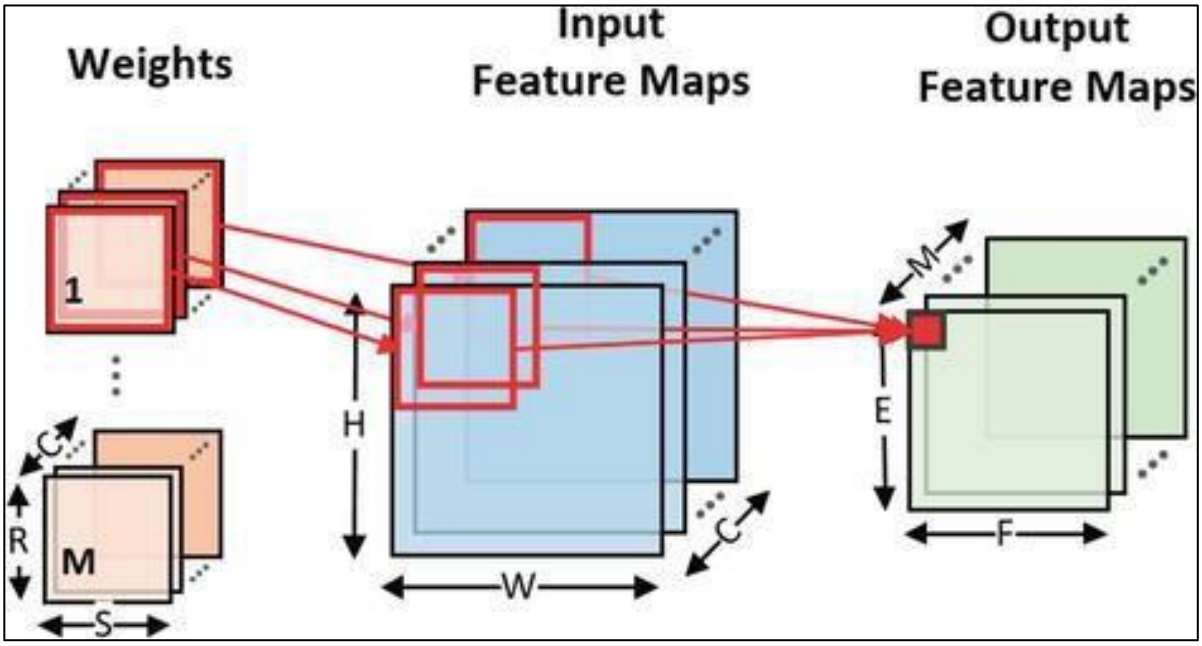

Figure 1.3: Convolution Layer.

1.6.2 Pooling Layer

Each feature map that passes through this layer—typically located between two convolution layers—is subjected to the pooling procedure. The pooling method allows for significant size reductions in photographs while maintaining the integrity of their essential elements. Regular cells are used to partition the signal, and the highest number within each cell is retained. Square cells that are relatively small in size are frequently used to avoid the loss of a great deal of data. The most common configuration has a space of 2 pixels between neighboring 2x2 cells or 3 pixels between neighboring 3x3 cells (thus overlapping). The number of feature maps we acquire is the same as the input, but the final product is much more manageable. The pooling layer streamlines the network's calculations and reduces the complexity of its parameters. The network's efficacy is improved, and over-learning is prevented as a result. After being pooled, the feature maps identify maximum values less accurately than the original feature maps. Identifying a dog, for example, requires only knowing where its ears would likely be located on its cranium [37].

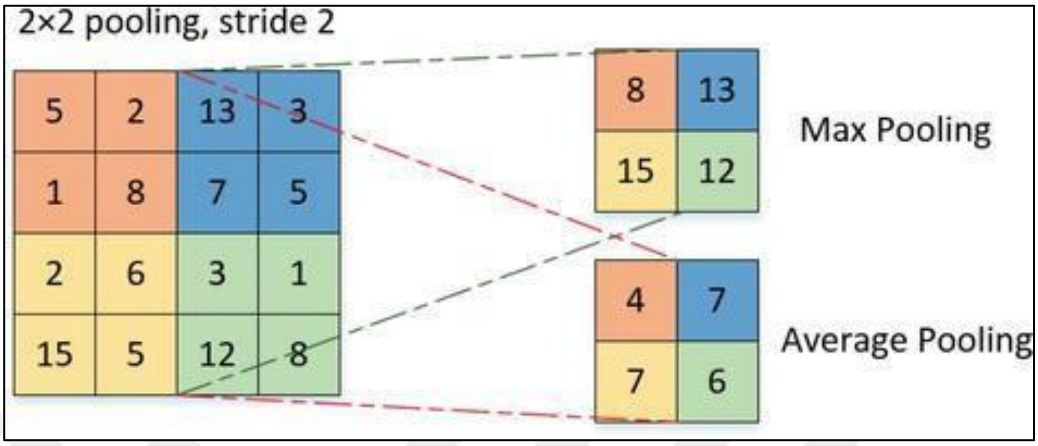

Figure 1.4: Maxpooling.

1.6.3 Fully Connected Layer

Whether the network is stacked or not, the fully connected layer is always the last layer in a neural network. This layer takes in a vector as input and outputs another vector. As a result, a linear combination and maybe an activation function are applied to the input data. The final, fully connected stratum is where the classification takes place. This layer generates an N-dimensional vector based on the number of categories in our picture classification assignment. Each element of the vector represents the likelihood that a picture falls into a particular category. The inputs are multiplied by their corresponding weights before being subjected to an activation function (logistic if N=2, softmax if N>2) that is applied to the aggregate of the weighted inputs. The weight is represented by multiplying the input vector by a corresponding matrix. The term "fully connected" is apt because every input value is associated with every exit value. The convolutional neural network learns the weights of its convolution layer filters through a process of backpropagation of the gradient during the training phase. The completely linked layer is responsible for assigning a category to a feature based on its position in the signal. Each high value in the input table indicates the location of a feature in the input signal (to varying degrees of precision depending on the pooling). When a feature's position in a signal is highly indicative of a specific object class, its value in the table is increased [36].

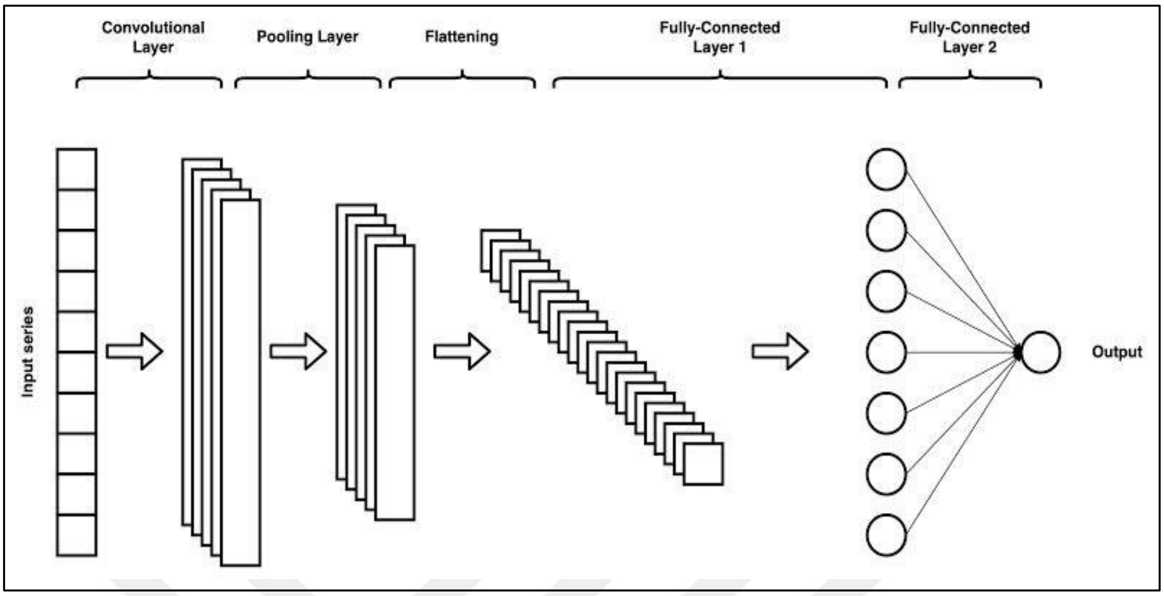

Figure 1.5: Fully Connected Layer.

1.7 REGULARIZATION

Regularization, in which a word is added to the objective function, is widely used to improve the model's generalizability. This parameter prevents the model from becoming too specific to the training data by ensuring that it ignores any random variation present in the data.
Loss Function (Error Term) + Regularization Term Equals Objective Function
Because of this, the objective function can be expressed as
In this case, The definition of the objective function is the sum of the loss function, L(F(xi),) in terms of the model's output F(xi) and the model's parameters, and the objective function, f(xi), in terms of the parameters. Regularization parameter and parameter norm f() make up the second word f() in the expression.
In CNN, two regularization techniques are used that are remarkably close to one used in linear regression:

Norm in L1: f() = ||||

The total number of model factors is 1.

Norm in L2 space: f() = ||||

All of the model factors have a square root of 2

This Is Why We Have to Do It All Regularized The variance is a feature of the classical least squares model. This model does poorly on data sets that were not included in its training collection. Regularization significantly reduces the model's variation while having no appreciable effect on its bias. In order to control the effect of bias and variation, the tuning parameter is used by the regularization techniques we've covered so far. As lambda grows, the coefficient values decrease, and the variance rises. This increase is helpful up to a point because it stops variance from being reduced too much (overfitting) without compromising any essential properties of the data. However, once the model passes a certain point, it begins to lose crucial details, which can lead to prejudice and underfitting. It is for this reason that &lambda's value should be carefully chosen [38].

1.8 BATCH NORMALIZATION

Let's define "normalization" before we dive into the Batch normalization process. Data normalization is a preprocessing technique for making numerical data consistent on a single scale without altering its original form. When feeding data into a machine learning or DL algorithm, we typically normalize the numbers to make the input more manageable. One of the main reasons we standardize is to guarantee that our model can be used in a variety of circumstances. In conclusion, to recap, Batch normalization is a method of improving the performance and stability of neural networks by augmenting them with additional levels of deep learning. The input of a layer is standardized and normalized by a new layer. When asked why group normalization uses the word "batch," Data is often collected and arranged into sets, or batches, in order to train a neural network. In the same way, batch normalization does not use a singular input but rather a series of batches to perform the normalization process.

Now that we know what Batch normalization is and why it's useful, let's look at how it operates. There are two stages involved. Before performing rescaling and offsetting, the data is normalized.

Normalization of the Input

When data is normalized, the mean is set to zero and the standard deviation is set to one. Here, we take the batch input from layer h and average out the concealed activation.

$$\mu = \frac{1}{m} \sum h_i \tag{1.1}$$

Here, m is the number of neurons at layer.

The next step is to determine the standard deviation of the concealed activations after we have meant at our end.

$$\sigma = \left[\frac{1}{m} \sum (h_i - \mu)^2 \right]^{1/2} \tag{1.2}$$

Now that we are aware of the meaning, we can also determine the standard deviation. We'll create a baseline for the concealed activations using these figures. To achieve this, we must first compute the smoothing term (), which is obtained by taking the standard deviation of each input and subtracting it from the value. Next, we must divide the resultant number by the sum of the standard deviations and the smoothing term.

By preventing division by a zero value, the smoothing term (ε) ensures numerical stability inside the operation [39].

$$h_{i(norm)} = \frac{(h_i - \mu)}{\sigma + \varepsilon} \tag{1.3}$$

1.9 DROPOUT

It is an organization method that prevents overfitting, as overfitting is something that occurs constantly in neural networks, and this affects the results obtained from training. For this reason, Geoffrey Hinton invented the dropout system in order to control overfitting and thus obtain better training results.As for how dropout works, initially during training, the dropout selects a random set of sub-neurons for each small set of data and sets their output to zero. This leads to the effective removal of these neurons from the neural network, after which the network becomes more powerful, durable, and less susceptible to overruns, as the network cannot rely on any feature or just one neuron. After that, when making predictions, this leakage is usually stopped, and then all neurons from the network are used. However, their output is measured according to the leakage rate that we obtained to ensure that these predicted results remain the same during training. [40].

1.10 DENSE LAYER

The dense layer is a feature in the Keras library that allows a CNN network to train faster by using an adaptive learning rate that decreases with the amount of training data.

The dense layer replaces an 8-bit fully connected network with a much more computationally expensive 16-bit fully connected network, which improves the computational performance of

A form of neural network called a fully connected network has connections between each unit and every other unit in the network.

"The dense layer is a very powerful part of the neural network toolbox for signal processing and computer vision applications.

Let's imagine you have a feed-forward multilayer perceptron with three layers - input, hidden and output.

a. A model is trained on many signals (for example, a billion) and uses convolution operations to define its features.

b. There are X inputs with corresponding values ranging from 0 to 1, and the input data is distributed over all the neurons in the first layer.

c. The network's weights are initialized randomly as weightiness and then the weights are iteratively updated through backpropagation until the convergence criterion is

Unlike a fully connected neural network in which the units have no constraints on their connectivity, a dense convolutional neural network has units with bounded connections, which is much more powerful.

1.11 CNN WITH ECG

CNN often collaborates with ECGs to produce more accurate news reports. ECGs provide CNN with detailed information on heart activity and can help the network to produce more accurate reports.

In recent years, heart monitoring equipment has become much more advanced. This has allowed doctors to monitor their patients' health and prevent any potential complications more effectively. For example, if a patient develops an irregular heartbeat or cardiac arrhythmia, a doctor can use an ECG to monitor the patient's pulse and determine the cause of the problem. By monitoring the heartbeat more closely, doctors can ensure that the patient receives the appropriate treatment quickly, which is especially important in the case of a medical emergency. By using ECGs in conjunction with CNN, doctors can provide the public with accurate and up-to-date information regarding the health and activity of the heart. ECGs are often used as a diagnostic tool to detect heart abnormalities. Doctors use an ECG to determine the cause of irregular heartbeats and other cardiovascular problems. They can also use ECGs to monitor the overall health of patients and to diagnose heart problems before they develop. If a patient has symptoms such as chest pain or shortness of breath, an ECG can be used to detect the underlying cause of the problem so that doctors can treat the condition before it worsens [41].

2. RELATED WORK

2.1 LETERETURE REVIEW

Shweta H. Jambukia [1] et al have presented a work to classify EKG data in which they present two different methods of classification. The first method was the classification of the heart rate signal, which is the more difficult method that few researchers have worked on compared to the classification of the ECG. The primary goal of this effort is to remove noise from classification and obtain results with good accuracy. Since waves and algorithms have been used in this work, such as the Pan-Tompkins algorithm, wave has been used for noise removal and feature extraction. The researchers also used SVM technology to classify the results here that neural networks are good for ECG classification in terms of accuracy compared to the training and testing dataset. To calculate this performance, to classification confusion matrix be used in this work. The MIT-BIH database that was used on the job. It is observed that MLPNN gives good results in terms of accuracy.

Abdul Rahman Pemankar [2] and others have proposed a deep learning approach called DENS-ECG. The project's goal is to identify the ECG waveform and the possibility of applying the algorithm from home of course by a cardiologist, which means that it is easy to deal with the discovery and classification of the EKG. As DENS-ECG combines CNM and LSTM, and the waveform is determined by T, QRS and P. This task is considered one of the most difficult in waveform determination. ECG sections were utilized to effectively train and test the model, and DENS-ECG was used to extract temporal characteristics from the 1D ECG signal and categorize the four classes NW, T, QRS, and P. 5 times the cross section to train this model. A model trained on secret and hidden test sets was used to assess the algorithm's performance model, and it performed well, with F199.56% and accuracy of 96.78% in the MITDB and QTDB data sets, respectively.

In this paper, Rajkumar. A, Ganesan. M, Lavanya. R, et al. create a cognitive-based, Deep Learning-based method for classifying electrocardiogram (ECG) signals (DL). A variety of cardiac conditions can be identified with the help of an electrocardiogram. The abnormalities in an electrocardiogram (ECG) are called rhythms, and they include conditions like atrial fibrillation, ventricular tachycardia, ventricular fibrillation, and so on. This project's main objective is to identify and describe the patient's numerous heart rhythms. The categorization of arrhythmia types using a deep learning system is encouraged by this investigation. They use a Convolutional Neural Network (CNN) DL approach in this scenario, which is excellent at categorizing signals. The MIT-BIH database's time-domain ECG signals are downloaded to Physiobank.com, where they are processed by CNN to extract recognized traits. Because to the particularly adapted feature, which takes the place of the manually derived characteristics in this study, cardiologists will be able to efficiently screen patients for heart disease. The MIT-BIH ECG dataset and the irregular heartbeat signal classes were utilized to train and assess the CNN. We alter the activation function and the number of training iterations in order to assess the efficacy of the suggested method. They found that enabling the ELU improved accuracy to 93.6% with a reduction of only 0.2%. [3]

Muhammad Yamako Mert Doman et al. proposed a low-cost, high-resolution ECG tracking device to be used with wearable, wireless sensors to detect arrhythmias in the patient. The pre-supervised techniques for ECG tracking require both abnormal and normal heartbeats to train the custom classifier. Only those with a known history of cardiac issues may access this data, which can be used to train a tailored algorithm built into a wearable device. This research (1) proposes using sparse dictionary learning to create a space with only good signals, and (2) then investigates if a straightforward blank space projection or a classification algorithm based on least squares may improve detection accuracy while lowering computing complexity. They (2) introduce a representation-based sparse field conditioning approach to train the custom classifier without exposing the new user to aberrant heartbeats by displaying both unusual and typical signals on the signal space for the new user to existing users. This means that non-impulsive learning can be achieved without resorting to methods such as inducing an irregular heartbeat. The strategy significantly outperforms earlier work when a field-adaptation based training data generator is paired with a straightforward CNN 1-D classifier, according to extensive testing on the common MIT-BIH ECG dataset. As a third step, the researchers suggest a group classifier that combines (1) and (2) to further improve performance (2). The F1 score for this method is 92.8%, and the mean accuracy is 98.2%, allowing us to detect no arrhythmias [4].

Jintai Chen and others propose a new manually optimized neural network, HRNN, for classifying arrhythmias. ECG-based diseases are the first class of diseases to be classified using both manual grammatical approaches and neural networks. Studies have shown that

HRNN outperforms the most recent techniques by a wide margin (e.g., more than 4% in total recall score and 13% in average recall score for each category) and has the highest overall recall rate Score and highest average recall score for each category in both sets of TianChi ECG data. The model who had the highest overall average in each area, F1, is one of the most recent trends. also experiments She clarified that HRNN may be used to spot false flags. Examples demonstrating strong promise for some realistic jobs in artificial intelligence-assisted masking restoration of damaged labels and annotations [5].

Likith Reddy and colleagues proposed an interpretable multi-level model for multi-channel ECG categorization. Results from an evaluation of the suggested model on the PTB-XL dataset show that it achieves better results than a number of competing models. For insight into how the model's behavior shifts when exposed to varying sets of ECG channels, an ablation study was conducted. The generated attention scores were displayed graphically, and then compared to the recommendation’s cardiologists use to identify the two subtypes of myocardial infarction. It is critical to be able to analyze the model's results in order to better understand how the suggested model may be helpful in clinical decision-making in practice. The model's prospective interpretability can be measured against various cardiac disorders. They recommend a model that makes use of the multi-channel data seen in the common 12-channel ECG recordings to learn patterns at the beat, rhythm, and channel levels. According to the experimental results, the model achieved the greatest F1 score on the PTB-XL dataset of 0.8057, the mean accuracy of 88.85%, and the macro-averaged ROC-AUC score of 0.9216. Comparisons are made between the interpretable model's attention visualization findings and the cardiologists' recommendations to ensure accuracy and practicality. [6]

One of the top ECG programs, the 12-lead ECG that Temesgen Mehari and Nils Strout [7] suggested, this form of vital signs is the most common. Public ECG data collections have grown in availability, but a lack of names is still a major obstacle. Potential benefits of self-supervised learning Answer to this predicament in the form of a plan of action. As a result, more intense workouts would be possible. Improved predictions for rare diseases using the same quantity of classified data by merging or expanding training data sets. The authors present the first extensive evaluation of unsupervised learning using 12-lead Electrocardiogram data from clinical practice. This is why they are modifying cutting-edge techniques from the area of self-supervision Discrimination, such as latent prediction of ECG readings. The first stage is to acquire knowledge of the various representations and assess its quality through a written evaluation of A freshly comprehensive and comprehensive ECG classification assignment. The second stage involves a comparison of supervised and unsupervised training for a few ECG classifiers. performance. The best performing approach, a modification of a cross-predictive encoder, was shown to only perform linear assessment 0.5% worse than supervised performance. As compared to supervised performance, label efficiency, and lifespan, exact models perform 1% better downstream. contrary to the physiological turmoil in the backdrop. This study conclusively demonstrates the viability of self-supervised learning for extracting discriminant representations from ECG data, as well as its many benefits on final tuning. When compared to fully skewed training, such representations are advantageous in downstream tasks.

It was reported by Linhai Ma and colleagues that in several research laboratories, DNNs have been created to automatically read ECG signals and find probable cardiac problems in human hearts. With a large enough amount of data, studies have demonstrated that DNN classification accuracy can rival that of cardiologists with human expertise. While DNNs have been shown to perform admirably in terms of classification accuracy, they are still susceptible to adversarial noises, which are small changes in the input of a DNN that result in an incorrect class-label forecast. Since ECG signal classification is a life-critical application, it is both difficult and necessary to increase DNNs' robustness against adversarial noises. They created a CNN for categorizing 12-lead ECG signals of different lengths, and then they employed three defensive strategies to strengthen the CNN for this classification task. This study's ECG data is difficult to work with due to the small sample size and wide variation in sampling duration. In accordance with the top six entries in the CPSC2018 ECG classification challenge, evaluation results show that the customized CNN attained a satisfactory F1 score and average accuracy, and that the defense methods improved the robustness of CNN against adversarial noises and white noises with only a minor decrease in accuracy on clean data. In this paper, a convolutional neural network (CNN) is created for the classification of ECG signal data from the 2018 China Physiological Signal Challenge that can handle high-dimensional and variable-length input. They used three typical protection strategies to make CNN more resistant to noise. Comparing the assessed defensive techniques to the model trained solely using cross-entropy loss and clean data, all of the defensive strategies improved CNN's resilience against 100-PGD and white noise. Contrary to conventional adversarial training, regularization-based defense strategies were found to be more robust to white noise. By analyzing how responsive adversarial training is to the user-specified noise level, this was found. [8]

Among others, Yunan Wu in this study, the classification performance of convolutional neural networks using one-dimensional signal input and two-dimensional picture input is compared. Swish activation in 1D-CNNs, which may be utilized for ECG classification, offers greater accuracy and resilience compared to other activation functions. They apply weights pretrained on signal Net to initiate the similar 2D-CNNs model after discovering that random initialization outperforms 1D-CNNs in terms of performance, successfully resolving the overfitting issue. This suggested method's accuracy is up to 98.00%. In contrast to conventional classification techniques, the signal input method used by 2D-CNNs does not need manually extracting features. With a big database for fine tuning, 2D-CNNs may achieve improved accuracy and resilience compared to 1D-CNNs' signal input approach. The comparison with cutting-edge techniques shows that the recommended approaches produce the best results. Also, the suggested strategy performs well on another classification with five classes and achieves an accuracy of 94.5%. In order to accomplish automated and real-time categorization, it may be possible to attempt inputting a portion of ECG data in the future rather than individual ECG beats. [9]

Muhammad Kachuee as well as others The majority of research, according to them, center on Instead of transferring knowledge and skills from one assignment to another, a group of circumstances are sorted into a data set that is noted for this mission. In this study, they propose an accurate heart rate categorization approach based on deep convolutional neural networks. AAMI EC57 Basic classifies arrhythmias into five main categories. Also, they propose a mode of transportation. This assignment's expertise for classifying myocardial infarction (MI) tasks. Using the MIT-BIH and PTB Diagnostics datasets from Physio Net, they assessed the suggested technique. According to the findings, the suggested technique can classify MI and arrhythmias with an average accuracy of 93.4% and 95.9%, respectively. [10]

In this study, to classify ECG signals into eight different categories—normal, premature ventricular contraction, paced, right bundle branch block, left bundle branch block, atrial premature contraction, ventricular flutter wave, and ventricular escape—Amin Ullah and colleagues have proposed a 2-D convolutional neural network (CNN) model. The original one-dimensional ECG time series data are transformed into two-dimensional spectrograms using the short-time Fourier transform. For effective feature extraction from input spectrograms, two-dimensional convolutional neural network (CNN) models with four convolutional layers and four pooling layers are used. Using the publicly available MIT-BIH arrhythmia dataset, we evaluate our methods. They managed a 99.11% average classification accuracy, which is higher than the findings of recently published work on classifying analogous arrhythmias. Other indices, such as sensitivity and specificity, show similarly promising results, demonstrating the efficacy of the proposed approach. [11]

Both Yusra Obeidat1 and Ali Mohammad Alqudah suggested to achieve these goals, For automated, quick, and precise ECG classification, this article uses a lightweight hybrid 1D deep learning model that blends Convolutional Neural Network (CNN) and Long Term Memory (LSTM) Techniques. The LSTM and CNN algorithms are both It was built independently to be put up against the CNN-LSTM model mashup. the quantity of parameters utilized, how long it takes to categorize, and how accurate it is. A hybrid CNN LSTM system can recognize and categorize various cardiac rhythms automatically, including normal sinus rhythm (NSR), atrial fibrillation (AFIB), atrial flutter (AFL), atrial premature beat (APB), left bundle branch block (LBBB), and right bundle branch block (RBBB). For ECG pulse selection and deep feature extraction, the hybrid model makes use of CNN blocks. As a result of training, the LSTM layer will become adept at gleaning temporal context information from documents. The findings show that the proposed hybrid CNN-LSTM model achieves high accuracy and sensitivity of 98.22% and 98.23%, respectively. This design is lightweight and When compared to other models that have been used in the past, it is superior in terms of speed and accuracy when categorizing ECG beats. Excellent for clinical application embedded system architecture. Improve the speed and accuracy of cardiac disease monitoring [12].

In this paper, Abu Hasnat Mohammad Rubaiyat and coauthors suggest a novel approach to the classification of 1D signals. The recommended approach employs a closest subspace search technique in the signed cumulative distribution transform (SCDT) space to produce a non-iterative solution to the classification issue. Several tests show that the proposed approach not only outperforms well-known CNN-based approaches but also is data-efficient and resistant to out-of-distribution samples. In addition, unlike competing techniques, it does not necessitate familiarity with signal class templates or data deformations to produce accurate results. We will look at effective ways to teach these general mathematical categories for the space of signal deformations G in the future. This attempt resulted in a 61% success rate and a 0.59 f1 error rate [13].

This study by Khiem H. Le and colleagues highlights the vulnerability of the cardiovascular system during the COVID-19 pandemic, which highlighted gaps in health care provision. Throughout the globe, there is a growing demand for innovative methods to provide services in a timely and cost-efficient manner. Checkup and prognosis. Clinical evidence suggests that COVID-19 infection can lead to heart damage, and the electrocardiogram (ECG) may be a useful indicator for detecting the presence of COVID-19. The goal of this study is to use ECG data to automatically detect COVID-19. They suggest a novel strategy for digitizing paper ECG data, which involves After that, it's used as input into a 1D-CNN for learning and disease detection. R achieves its maximum value when used to assess the superiority of a digital signal at. Printed copies of electrocardiograms have been labelled. After determining the RR intervals for each digital signal, the results were compared to the matching signal. Experiments with COVID-19 ECG signals show the effectiveness of the recommended digitization method the method successfully recovers the genuine signals with a mean absolute error of 28.11 Ms. Individuals infected with COVID-19 can be distinguished from non-infected people using a model 1D-CNN (SEResNet18) trained on digitized ECG Signals. Both COVID-19 compared to normal and COVID-19 versus other groups show an accuracy of 98.50% in their respective classifications. The suggested method also performs exceptionally well on the multiple classification job. The findings point to the viability of using a DL system taught on digital ECG signals in When used as a possible method for identifying COVID-19 [14].

And others, including Ertan Butun: The use of deep learning-based methods in computer-aided diagnosis tools has recently increased in popularity. One of the most interesting recent developments in deep learning is the use of capsule networks. In this research, we utilized the 1D version of CapsNet to automatically identify CAD in two-second (95,300) and five-second (38,120) ECG segments. These samples come from 40 healthy people and 7 with coronary artery disease. In experimental research, the 5-fold cross validation approach is employed to assess the model's efficacy. In terms of 5-fold diagnostic accuracy, the proposed model, known as 1D-CADCapsNet, demonstrated good results, with 99.44% and 98.62% for two- and five-second ECG signal groups, respectively. Our results employing a 2-second ECG segment are the most effective ones to date when compared to cutting-edge studies mentioned in the literature. The 1D-CADCapsNet model autonomously learns the pertinent representations from raw ECG data and may be utilized as a quick and accurate diagnostic tool for cardiologists without the need for any manual training. Here, researchers go into great detail[15].

Including Md Abid Hasan and Others The authors of this article use convolution to the neural network (CNN) was used to categorize people's heart rates into five different groups, and the generated taxonomy output was used to incorporate data into the smart heart monitoring system (SHMS), which consists of a smartphone app with some user-friendly functions for patients. At Physio Net, we tested the algorithm with a variety of data sets. It was found that the algorithm had an average accuracy of 98.28% when classifying five distinct types of heart beats. With this effort, we hope to create permanent circumstances where Human ECG can thrive. When the heart's normal beat is disrupted, a number of potentially life-threatening arrhythmias can develop. Electrocardiograms (ECGs) are a trusted method for monitoring heart and blood vessel function. In modern times, it has come to be recognized that pinpointing the precise classification of the heartbeat is a highly stressful accident [16].

In this paper, Ayman Rabee and Imad Barhumi propose a highly trustworthy approach for evaluating and categorizing electrocardiograms using discrete wavelet transform multiresolution analysis and support vector machine classification (SVM). This method has three parts: preprocessing the ECG data, choosing features, and identifying heart rhythms. The wavelet transform is used to preprocess and denoise the signal, and then to use the coefficients of the transform as features for each ECG pulse as inputs to the classifier. The incoming ECG beat is classified using a classifier built with support vector machines. From the MIT-BIH arrhythmia database, 17260 ECG beats were chosen for this study. Classification accuracy for the 14 different kinds of heartbeats is a mean of 99.2% [17].

Giorgos Giannakakis and the other authors of this paper propose a Deep Learning multi-kernel architecture for categorizing stress stages based on heart activity. The results of related DL studies (achieving 90% in [26] and 73-87.4% in [28]) and the outcomes of a related ML approach using the same dataset (84% in [25]) are superior to those of the proposed DL methodology, which, when evaluated using 6-fold cross-validation, can achieve a classification accuracy of up to 99.1%. As far as we can tell, the proposed method is the first to employ a multichannel 1-dimensional CNN in the DL literature; all other similar works have relied on single-kernel CNN. With the help of the recommended technique, a reliable experimental procedure for data-driven and learning-based models is presented, and the function of various signal transformations for various stress types is disclosed. Given the experiment's short sample size, several models were evaluated using sliding window analysis and 6-fold cross-validation. Another significant drawback of a data-driven approach like this one is the inter-subject variation in the many categories of stress (experiment phases). To deal with this, separate models were developed for the four distinct stresses. The model's credibility would rise with more study and a bigger sample size, which would then result in parameter optimization, higher performance, and more general models for usage in testing environments. [18].

In the past, acquiring physiological signals like an electrocardiogram (ECG) for extended periods of time was prohibitively expensive, but with the advent of inexpensive wearable Internet of Things (IoT) devices, this is no longer the case. Automatic analysis methods are crucial for mining useful insights from this mountain of data. From electrocardiogram (ECG) signals captured by a wearable device, in this article, a 1-dimensional convolutional neural network (CNN) for classifying heartbeats is introduced. The method suggested by Li Xiaolin and colleagues has been tested with the Physionet MIT-BIH Arrhythmia database, and it successfully divides heartbeats into the five categories required by the AAMI standard. We used the SMOTE method to enrich the dataset to correct for the disparity in the number of classes present. Following training on the enhanced data, the network displayed 98.12% accuracy, 98.07% sensitivity, and 98.29% specificity [19].

Cardiovascular diseases (CVDs) are now the main cause of mortality worldwide due to their increasing prevalence. Getting an electrocardiogram is the gold standard for identifying cardiac problems (ECG). Traditional 12-lead electrocardiograms are still used in many modern scientific and clinical uses. Nevertheless, ECG may be incorporated with portable or wearable devices, increasing its accessibility, if the number of leads is reduced. Two novel techniques are presented in this paper by Khiem H. Le and colleagues to raise the existing deep learning system's capability for classifying 3-lead ECGs to the level of models trained on conventional 12-lead ECGs. We propose an efficient mechanism for incorporating patient demographic data into the system, as well as a multi-task learning approach implemented as a regression on the number of heartbeats. They achieved F1 scores of 0.9796 and 0.8140 on two sizable ECG datasets (Chapman and CPSC-2018), respectively, outperforming even the most advanced ECG classification algorithms trained on 12-lead data [20].

There were a few others, but Ali Mohammed was one of them. Utilizing a compact 1D hybrid deep learning model that blends long short-term memory (LSTM) and convolutional neural network (CNN) techniques, this research automatically classifies beats in an electrocardiogram (ECG). To help with the creation of the hybrid CNN-LSTM model, we analyzed the accuracy, quantity of factors, and classification time of the solo CNN and LSTM models. Normal Sinus Rhythm (NSR), Atrial Fibrillation (AFIB), Atrial Flutter (AFL), Atrial Premature Beat (APB), Left Bundle Branch Block (LBBB), and Right Bundle Branch Block (RBBB) are just a few examples of abnormal heart rhythms that can be automatically extracted and classified using deep features from the combined CNNLSTM system (RBBB). The ECG rhythm is analyzed using a hybrid model that makes use of the CNN blocks to select and retrieve deep features. While that is happening, the LSTM layer will be taught to extract important temporal context. The results demonstrate the proposed hybrid CNN-LSTM model's excellent accuracy and sensitivity, which are measured at values of 98.22% and 98.23%, respectively. When compared to previous models used for ECG beat classification, this one is better. Also, it is incredibly lightweight, which makes it ideal for embedded system designs that may be applied in clinical settings for quicker, more effective monitoring of cardiac problems [21].

Similar to the method suggested by Thao Nguyen, we present a brand-new and incredibly effective method in this article for automatically recognizing COVID-19 from ECG printouts. The proposed technique converts digitalized ECG printouts into 1D ECG signals using a signal-to-signal conversion algorithm, which can then be used to train a 1D SEResNet18 model. The findings of these tests in a variety of contexts have been analyzed.

To more accurately assess the worth of the collected signal, we also created classification models by mimicking the data configuration of earlier research. We then compared our findings to their findings. Data analysis shows that the proposed method outperforms previous work with the same dataset, indicating that the ECG signal may be used to categorize COVID-19 patients. The proposed method of differentiating COVID-19 ECG might potentially be used as proof that COVID-19 patients experience certain alterations in their ECG signals. Future research can readily convert the recommended ECG-based COVID-19 diagnostic to work with real-time systems and mobile applications. Through further interpretation of the model's decision, researchers expect to understand how COVID-19 patients' ECGs vary from those of healthy individuals. This has the potential to benefit the medical community by providing a more rapid and accurate means of detecting COVID-19, as well as by reducing hospital expenses by reducing the frequency with which patients need to be seen. The success percentage of their final algorithm is 98.45 percent [22].

In an effort to automate the detection of coronary artery disease (CAD) using electrocardiogram (ECG) data, researchers Ertan Butun and colleagues[23] are employing capsule networks (CapsNet). Recently, there has been a rise in the popularity of using computer-assisted diagnosis tools that employ techniques based on deep learning. The implementation of capsule networks is one of the most exciting new trends in deep learning. We used the 1D version of CapsNet in this study to autonomously detect CAD in two-second (95,300) and five-second (38,120) ECG recordings. Forty healthy individuals and seven individuals with coronary artery disease contributed to these data. Scientific research uses the 5-fold cross validation method to evaluate the effectiveness of the model. In terms of 5-fold diagnosis accuracy, with performance scores of 99.44% and 98.62% for the two-second and five-second ECG signal groups, respectively, the suggested model, known as 1D-CADCapsNet, performed superbly. Our results using a 2-second ECG segment are the most up-to-date and comprehensive of any similar studies mentioned in the literature. Cardiologists may use the 1D-CADCapsNet model as a quick and precise diagnostic tool without having to engage in a lot of time-consuming manual labor since it automatically learns the relevant representations from raw ECG data. [23]

In this paper, Hari Mohan Rai suggested a method to analyze ECG data using discrete wavelet transform and morphology features. According to the types of ECG beats present in each category, records from the MIT-BIH arrhythmia database have been divided into normal and pathological groups. The normal class is determined to be twenty-five records, the abnormal class is twenty records, and the remaining three records are deleted. That's why this investigation employs a total of 45 records and 64 characteristics to make the signal classifications. We were able to improve the accuracy of the Back Propagation Network (BPN) and the Feed Forward Network (FFN) to 97.8 percent and 100 percent, respectively, by altering the number of neurons in the concealed layer, while the Multilayered Perceptron maintained 100 percent accuracy under all conditions. Results for training and testing samples of data using any classifier show the suggested work to be extremely effective, with most results exceeding 90%[24].

Accurate and exact delineation of QRS-complexes is essential to the success of computer-aided ECG analysis. The K-Nearest Neighbor (KNN) technique is suggested by Indu Saini and colleagues in this paper as a classifier for determining if a QRS-complex is present in an electrocardiogram (ECG). The proposed technique is evaluated on two manually annotated standard databases, CSE and the MIT-BIH Arrhythmia database. In this work, a digital band-pass filter is employed to reduce false detection brought on by interference in the ECG signal in addition to the gradient of the signal being used as a feature for QRS-detection. Further, the value of K and the distance measure chosen to have a significant impact on the performance of a KNN-based classifier. Five-fold cross-validation suggests that the optimal number of K for the KNN classifier is 3, and that the distance metric should be Euclidean. Using the CSE database, we were able to obtain a detection rate of 99.89%, while using the MIT-BIH database, we were able to achieve a detection rate of 99. The detection rates of the QRS detector were 99.86% sensitive and 99.86% specific for the CSE database, and 99.81% sensitive and 99.86% specific for the MIT-BIH Arrhythmia database, respectively. Using the CSE and MIT-BIH Arrhythmia databases, the authors compare and contrast their suggested algorithm with previous published work. With these findings, the KNN algorithm has been established as a solid option for accurate and dependable QRS-detection[25].

It was suggested that G. Rajender Naik and others In order to properly categorize an ECG trace, the technique of Multimodal Decision Learning was introduced. The findings demonstrate that, compared to the ANFIS approach, the most effective adaptive neuro-fuzzy algorithm, the suggested model offers superior categorization. The percentage of accurate diagnoses, the quantity of false positives, the quantity of false negatives, the false rejection ratio, the false acceptance ratio, the global acceptance ratio, the confusion matrix, the Kappa coefficient, the sensitivity, the specificity, and the accuracy were just a few of the metrics used to present the results. Records from the MIT-BIH database were used for the evaluation. The effectiveness of the performance evaluation of the given method may be further evaluated by comparing the results with those of well-established techniques like support vector machines and artificial neural networks. This attempt has an accuracy level of 94.2% [27].

This paper's suggested method is based on previous work by C. VENKATESAN and colleagues, specifically their work on feature extraction and classification. Adaptive filters based on the DENLMS algorithm are used to preprocess electrocardiogram (ECG) signals to improve filtering efficiency while keeping computational complexity to a minimum. Because Coiflet wavelet collects all potential R-peaks and provides a more accurate beat rate, it is utilized to detect R-peaks. An SVM classifier, which is simpler to develop than other machine learning methods, is used to categorize arrhythmic beats using the acquired beat rate and HRV Frequency domain parameters. Parameters like electrocardiogram (ECG) and heart rate variability (HRV) data are used in conjunction with PCA, ANN, knowledge-based system, KNN, and SVM categorization methods. Using these methods, we were able to obtain a maximum accuracy of 94.2% in our classifications. However, when comparing normal and arrhythmia-risk abnormal subjects, the SVM-based classifier achieves a maximal accuracy of 96% in the experiments [26].

3. METHOD AND METHODOLOGY

3.1 1D CNN MODEL

A deep learning model type that has gained prominence recently is one-dimensional CNNs. These networks are essentially a set of stacked layers that use filters to extract features from data. The first layer in this type of network is a convolutional layer, which applies a filter to the input data to extract features. Each subsequent layer in the network uses the results of the preceding layer as its input and applies a new filter to these features to generate an output. The entire network is then trained using an algorithm called a backpropagation algorithm. In traditional machine learning methods, the training data is fed into the computer as input and the output are later computed. In contrast, a deep learning model uses a multi-layered architecture that extracts features from the input data and then predicts the output data based on the extracted features. This allows a model to make predictions about new data that it has not seen before without any additional training. Another unique feature of deep learning model is that they can learn over time based on experience rather than requiring the user to provide explicit instructions on how to make a prediction.

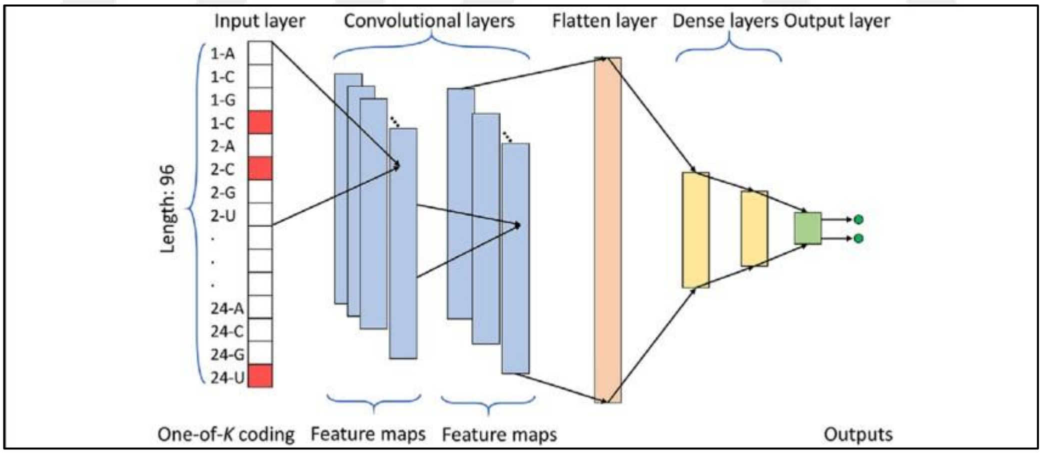

Figure 3.1: Layers of CNN.

3.2 DATASET

The Massachusetts Institute of Technology (MIT) and Brigham and Women's Hospital in Boston, Massachusetts, gathered the 2450 cardiac arrhythmias included in the MIT-BIH arrhythmia dataset from 2009 to 2013. It is one of the largest datasets of cardiac arrhythmias that are publicly accessible. The MIT-BIH arrhythmia dataset consists of electrocardiographic recordings from 2008 to 2012 performed on over 3700 patients (<5 years of age or >70 years) who had implantable cardioverter defibrillators (ICDs) implanted during routine medical care procedures. The data were collected using a dedicated computer system and stored as raw files by the hospital on magnetic tapes. These raw data were later digitized and processed using custom-written software to create specific formats suitable for distribution to researchers. A dedicated database was created to manage these records and a web interface was created to allow access to the data through the Internet. This system also collects information about the type of device that was used to record each recording, the patient's name, date of birth, sex, race, type of arrhythmia, electrocardiogram leads used for recording, and various parameters related to the ECG signal such as heart rate, ventricular diameter, and PR interval.

The signals in the dataset match the electrocardiogram's shapes (ECG). For instances where the heart is damaged by myocardial infarction, various heartbeats, and normal heartbeats. Each section of these pre-processed, segmented data corresponds to a heartbeat. Two CSV files with training samples and testing samples each are included in the dataset. There are 87,554 samples in the csv. train file.

In the training and test files, each model has 187 input characteristics and one label that denotes the column categorization. The dataset's heartbeats are split into the following five groups:

a. non-ectopic beats (normal beat)

b. ectopic supraventricular beats

c. Ectopic Ventricular beats

d. Fusion beats

e. Unknown beats U

3.3 ACTIVATION FUNCTION

After the inputs are weighted, they are combined and fed into an activation function, sometimes referred to as a transfer operation. An activation function is a simple mapping between the input weight total and the output of the neuron. An activation function controls the level at which a neuron is activated and the strength of the output signal. In the past, neurons used basic step activation functions in which they would only output a value of 1.0 if the summed input was greater than a certain threshold, say 0.5. Typically, activation functions that are not linear are employed. Thus, the network is able to combine the inputs in more sophisticated ways, expanding the range of functions it can simulate. The s-shaped distributions of the logistic function (also known as the sigmoid function) and the hyperbolic tangent function (Tanh), both of which return values between -1 and 1, were used as examples of nonlinear functions. In recent times, the rectifier activation function has been proven to yield superior outcomes [42].

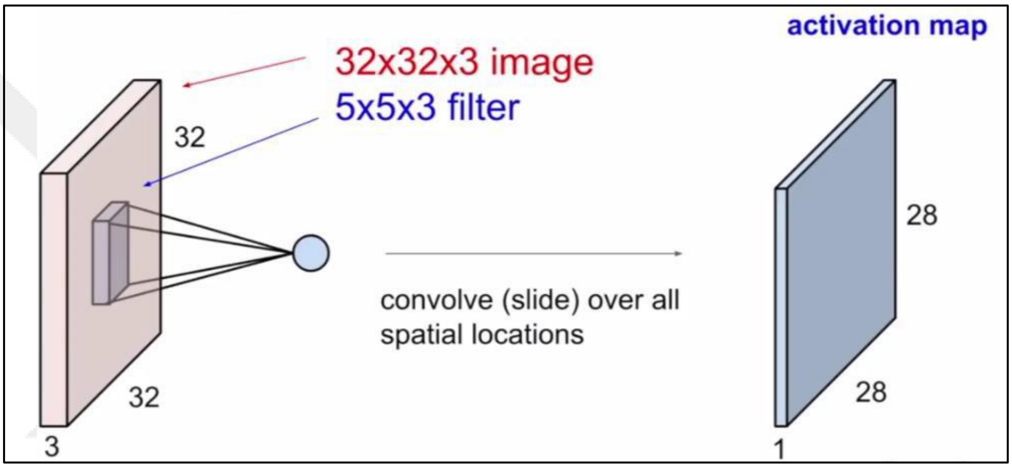

Figure 3.2: Activation Function.

3.3.1 Rectified Linear Activation Function (Relu)

The rectified linear activation function, often known as Relu, is a nonlinear or piecewise linear function that immediately returns the input if it is positive and 0 otherwise. Most neural networks use this activation function, and CNNs and MLPs in particular rely heavily on it.
Advantages of ReLU:

Since using Sigmoid or tanh in the hidden layers results in the dreaded "Vanishing Gradient" issue, ReLU is used instead. When the network is backpropagating, the "Vanishing Gradient" stops the previous layers from acquiring useful information. Since the output of a sigmoid function can only be between 0 and 1, it is best used in issues involving regression or binary classification and only in the output layer. As a result of saturation, the Sigmoid and tanh become less sensitive. There are many benefits to using ReLU, but here are a few: Calculation Made Easier: By keeping the derivative fixed at 1, as it would be for a positive input, we can increase the learning rate and minimize mistakes in the model. It has the property of representational sparsity, meaning that it can produce a valid zero.

Optimization and continuity are both simplified with linear activation functions. This means it performs best on supervised tasks with huge amounts of labeled data.

Disadvantages of ReLU:

When the gradient accumulates, an “exploding gradient" happens, leading to a drastic change in the subsequent weight updates. As a consequence, this leads to instability during learning and when convergent to global minima.

The "dead neuron" issue in ReLU arises when a neuron gets stuck in the negative feedback loop, where it always outputs zero. It's highly improbable that the neuron will make a full recovery if the gradient is 0. This takes place when either the learning rate is excessively fast or there is a substantial amount of unfavorable bias.[43].

3.4 FLATTEN LAYER

The flatten layer is a component of the CNN, which is a deep learning algorithm used for recognizing objects in digital signals. The flatten layer takes a series of feature maps produced by the convolution layers and flattens them out into a single vector that represents the features of a signal. This helps the network learn the differences between objects more accurately because it can analyze all the features at once rather than one feature at a time. Because the convolution layer is a fully connected network that performs a series of operations on each individual input signal to create multiple output signals, it can be difficult to determine which features are most useful for classifying a signal as a particular object. Since the flatten layer uses a vector to represent all of the features in a signal, it makes it much easier to analyze which features are most important by identifying which ones are most common in different signals. The process of flattening the input maps from each layer into the flatten map is called the pooling operation. Pooling refers to the process of taking an entire section of an output map and reducing it to a single value to represent the characteristics of the section[44].

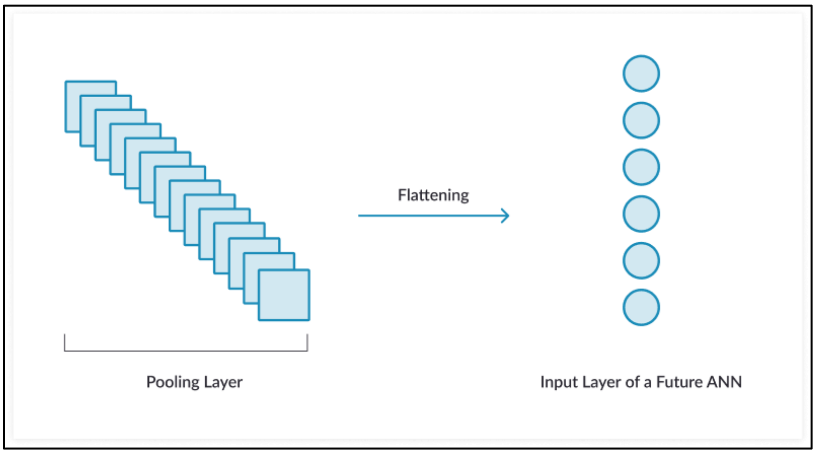

Figure 3.3: Flatten Layer.

3.5 SOFTMAX

While it is a subset of the sigmoid function, the softmax function comes in helpful when dealing with multi-class classification issues. Used frequently when managing several classes. In the output layer of signal classification issues, the softmax function was often present. The softmax function would divide the total of the outputs and squeeze the outputs for each class between 0 and 1. The softmax function is ideally used in the output layer of the classifier where we are acttrying attain the probabilities to define the class of each input. The softmax function is a popular model predictive inference algorithm used in machine learning and data preprocessing. The softmax function calculates the probability of a given output vector according to a given input vector. Let \(x_i\) represent an input vector of size \(n\) and let \(y_{i}\) be the corresponding output vector of size \(n\). The probability of the output vector \(y_i\) can be calculated as follows: \[\begin{align} p(y_i=k) = \frac{exp({w_k}^T x_i)}{\sum_{k=1}^{K} exp({w_k}^T x_i)} \end{align}\] where w~k represents the weights corresponding to each class \(k\), and K is the total number of classes. By using stochastic gradient descent with minibatch learning we can optimize the objective function which is given by the following expression: \[\begin{align} -\frac{1}{M}\sum_{i=1}^M l(x_i,y_i) \end{align}\] where \(l(x_i,y_i)\) is the loss function. In order to minimize this loss function, we use an optimization algorithm called SGD that works by repeatedly running the following steps:

Calculate gradients using the backpropagation algorithm with respect to the variables \(\theta\) of the neural network. Update the parameters using a step size determined by the learning rate. Repeat steps 2 and 3 until the error function is within an acceptable range or a predefined number of iterations have been reached.

3.6 PYTHON

Python is a complete computer language that lets you reuse your development libraries and code in your live models. Python, unlike Java, has a professional-level machine framework called scikit-learn and a stack of modules called SciPy for scientific computing. Theano, created at the University of Montreal, and TensorFlow, created by Google, are two of the best numerical tools for creating deep learning models. They share the fact that they were both designed to work with the Python programming language and that they can be used with the Keras library's incredibly intuitive API. To create our own deep learning and neural network models, we'll be using Keras, which hides the numeric processing complexity of Theano and TensorFlow behind a simple API. While at work, you'll build your own neural network and deep learning models from scratch.[28].

3.7 KERAS

Keras is a lightweight Python deep learning library that can be launched atop either Theano or TensorFlow. The goal of its creation was to streamline the process of creating deep learning models for scientific study. Given the underlying frameworks, it operates without a hitch on GPUs and CPUs and supports Python 2.7 and 3.5. It's distributed with the open-source MIT license. Francois Chollet, a Google engineer, created and continues to manage Keras according to these four tenets:

A model's modularity lies in the fact that it can be reduced to a series or a graph and still make sense. All a deep learning model's worries are standalone pieces that can be pieced together in any manner. The collection is minimalist because it contains only the bare minimum of resources necessary to accomplish the stated goal, with a focus on ease of use and clarity of presentation.

The system is designed to be easily extended by the addition of new components, allowing programmers to try out and experiment with different approaches to solving problems.

No unique file formats for Python's model files. The entire universe consists of local species. Python.[29].

3.8 OPTIMIZATION

Optimization is a mathematical system that determines the best solution in a quantitative sense. Optimization theory provides algorithms that help improve solutions or solve existing problems. Machine learning methods, and in particular neural networks, seek to minimize a loss or cost function, which measures how much actual data deviates from what would be predicted. The adam optimizer we used in this project is a hybrid between the RMS prop and the momentum optimizer; like RMSprop, It scales the learning rate parameters using the squared gradient and, like momentum, works by averaging across recurrent gradients. The program calculates various values based on the input. The momentum approach uses the accumulated gradient from previous steps in addition to the current step's gradient to find local minimums. To get things done, we combine SGD with momentum, which speeds up the learning process.

$$
\begin{aligned}
v_t &= \gamma v_{t-1} + \eta \nabla_\theta J(\theta) \\
\theta &= \theta - v_t
\end{aligned}
\tag{3.1}
$$

Momentum term $\gamma = 0.9$

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2
\end{aligned}
\tag{3.2}
$$

Here, beta 1 = 0.9 and beta 2 = 0.999 are the sole parameters utilized in Adam optimization, while m and v are moving averages of the gradients. g is the gradients in the mini batch [30].

3.9 OUR PROPOSED MODEL

Three pairs of MaxPool1D-Convolution1D layers are contained in the model. The values are reduced to 1D using a flatten layer. Three thick layers—two of which have an activation function—follow this. ReLU, and the final layer has 5 nodes with Softmax activation function, each of which corresponds to one of the five output class labels. When the output provided to the training is an encoded one hot, the softmax activation function is utilized. This function turns a vector of n values into a vector with n values up to 1, allowing the likelihood of each class being represented by n values to be calculated.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
re_lu_10 (ReLU)              (None, 265)               0
_________________________________________________________________
batch_normalization_9 (Batch (None, 265)               1060
_________________________________________________________________
dropout_4 (Dropout)          (None, 265)               0
_________________________________________________________________
dense_6 (Dense)              (None, 198)               52668
_________________________________________________________________
re_lu_11 (ReLU)              (None, 198)               0
_________________________________________________________________
dropout_5 (Dropout)          (None, 198)               0
_________________________________________________________________
batch_normalization_10 (Batc (None, 198)               792
_________________________________________________________________
dense_7 (Dense)              (None, 112)               22288
_________________________________________________________________
re_lu_12 (ReLU)              (None, 112)               0
_________________________________________________________________
dropout_6 (Dropout)          (None, 112)               0
_________________________________________________________________
batch_normalization_11 (Batc (None, 112)               448
_________________________________________________________________
dense_8 (Dense)              (None, 92)                10396
_________________________________________________________________
re_lu_13 (ReLU)              (None, 92)                0
_________________________________________________________________
dropout_7 (Dropout)          (None, 92)                0
_________________________________________________________________
dense_9 (Dense)              (None, 4)                 372
=================================================================
Total params: 416,641
Trainable params: 415,267
Non-trainable params: 1,374
_________________________________________________________________
```

4. RESULTS AND DISCUSSION

4.1 RESULT

When we have this type of business, the most important things that have a great impact on the results and on improving accuracy are:

a. Diagnosis

b. weight initialization

c. Learning rate

d. Activation functions

e. Network topology

f. Batch and epochs

g. Regularization

h. Optimization and loss

i. Early stopping

All these points through which we can improve the accuracy of the work and the results that we want to obtain. Then we can control the degrees of dropout, maxpooling and regpar, and this in turn can also control the results amazingly.

In the first table, we show how dropout can affect accuracy and results in general, because before dropout existed, deep learning had a huge overfitting problem. When the leak appeared it amazingly overcame that problem and so we could get much better results with it. When we changed the degrees of dropout, we noticed that the lower its score, the better the result we got, as shown in the table:

Table 4.1: A Table Showing the Effect of Drop Out on the Results.

Drop out	Loss	Accuracy
0.8 0.9	1.05 1.17	0.80 0.77
0.75	1.00	0.81
0.7	0.99	0.83
0.63	0.93	0.84
0.15	0.5	0.88
0.19	0.5	0.89

Now we turn to the effect of max pooling on our results. Pooling in general is basically a "zoom out" of the signal obtained from the previous layer. It can be compared to shrinking a signal to reduce pixel density. For maxpooling in general, the maxpooling layer and its work is very important and essential.

Because it reduces the resolution of the output given to the convolutional layer, the network will consider larger regions of the signal simultaneously from now on, this decreases the number of network parameters and hence the computational burden.

In addition, maxpooling may also help reduce overfitting.

In the table below, we explained how the maxpooling layer affects the accuracy of work, how it can increase accuracy and reduce loss. We noticed that when we reduced the rate of the maxpooling layer, it gave us better results.

Table 4.2: Table Showing the Effect of Maxpooling on Results.

Maxpooling 1d	Loss	Accuracy
16,8	0.55	0.87
14,6	0.5	0.88
13,5	0.5	0.88
13,8	0.5	0.89

Now we will talk about a very important factor that caused a very big change in the results, which is regpar. The regpar is an abbreviation for Receptive Field Peeling. It is a method used by the convolutional neural network when training on signals or data that have a lot of noise. The function of regpar is to reduce this noise and filter signals from any There are impurities in it in order to obtain a better accuracy of this signal and also to remove any part that does not belong to the basic signal parameters, as the basic function of the convolutional neural network is to recognize the real patterns of signals as recognized by the human , the regpar is a very important factor in order to improve Precision . The table below shows the effect of regpar on the results very significantly and obtaining much better accuracy.

Table 4.3: It Shows the Effects of Regpar and Maxpooling Together on the Results.

Regpart 1	Regpart2	Max pooling 1	Max pooling2	Max pooling3	loss	accuracy
0.002	002	31	13	8	0.5	0.88
0.0001	.0001	31	13	8	0.5	0.93
0.00001	.00001	31	13	8	0.3	0.94
0.00001	.00001	32	14	9	0.4	0.92
0.00001	.00001	10	10	7	0.2	0.98

Compared to the rest of the task, our accuracy was 98% and loss was 0.2%. We compared our work to another using the SVM classifier, which means for Support Vector Machines. It's an interesting algorithm with simple ideas. Using the hyperplane with the most margin, the classifier divides the data points. This is why the SVM classifier is called the feature classifier. SVM gets the most that can classify new data. Support vector machines are generally used for categorization, but they can also be used for regression. It simply handles many continuous and categorical variables. SVM creates a multidimensional super plane to divide classes. To limit error, SVM iteratively generates the super optimal level. SVM finds the dataset's maximal marginal hyperextension (MMH) to classify it. The model's 96% accuracy was lower than ours, as shown in the chart below.

Table 4.4: Comparison Table with Our Research.

Method Signal	Method signal	Classification Accuracy
ANN & PCA [24] &NCA [25]	ECG	88.5
KNN classifier [27]	HRV	90.4
SVM Classifier	HRV	96
Fuzzy KNN	RNN	92.5
Our proposed method	ECG	98

In the previous table, we compared our work with other work. The researchers used different algorithms than the algorithm that we applied in our work, and we concluded from the results that the results we obtained are much better in terms of accuracy.

4.2 TRAINING

```
Train on 5910 samples, validate on 2533 samples
Epoch 1/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 1.1063 - accuracy: 0.5963 - val_loss: 1.0798 - val_accura
cy: 0.6218
Epoch 2/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.8637 - accuracy: 0.7014 - val_loss: 1.0031 - val_accura
cy: 0.6656
Epoch 3/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.7759 - accuracy: 0.7445 - val_loss: 1.0239 - val_accura
cy: 0.6443
Epoch 4/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.7157 - accuracy: 0.7716 - val_loss: 0.8580 - val_accura
cy: 0.7134
Epoch 5/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.6614 - accuracy: 0.8034 - val_loss: 0.9839 - val_accura
cy: 0.7106
Epoch 6/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.6194 - accuracy: 0.8184 - val_loss: 0.8785 - val_accura
cy: 0.7284
```

```
Epoch 7/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.5397 - accuracy: 0.8547 - val_loss: 0.9027 - val_accura
cy: 0.7252
Epoch 8/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.5036 - accuracy: 0.8701 - val_loss: 0.9613 - val_accura
cy: 0.7201
Epoch 9/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.4427 - accuracy: 0.8910 - val_loss: 1.1214 - val_accura
cy: 0.6932
Epoch 10/25
5910/5910 [==============================] - 17s 3ms/sample - loss: 0.4306 - accuracy: 0.8978 - val_loss: 1.1791 - val_accura
cy: 0.7071
Epoch 11/25
5910/5910 [==============================] - 17s 3ms/sample - loss: 0.3948 - accuracy: 0.9096 - val_loss: 1.0847 - val_accura
cy: 0.7138
Epoch 12/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.3680 - accuracy: 0.9289 - val_loss: 1.2069 - val_accura
cy: 0.6857
```

```
Epoch 13/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.3431 - accuracy: 0.9352 - val_loss: 1.2650 - val_accura
cy: 0.7110
Epoch 14/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.3148 - accuracy: 0.9448 - val_loss: 1.2510 - val_accura
cy: 0.7355
Epoch 15/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2990 - accuracy: 0.9548 - val_loss: 1.3211 - val_accura
cy: 0.7233
Epoch 16/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2834 - accuracy: 0.9570 - val_loss: 1.4431 - val_accura
cy: 0.7233
Epoch 17/25
5910/5910 [==============================] - 20s 3ms/sample - loss: 0.2638 - accuracy: 0.9663 - val_loss: 1.4352 - val_accura
cy: 0.7114
Epoch 18/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.2900 - accuracy: 0.9569 - val_loss: 1.4079 - val_accura
cy: 0.7043
```

```
Epoch 19/25
5910/5910 [==============================] - 19s 3ms/sample - loss: 0.2929 - accuracy: 0.9565 - val_loss: 1.3820 - val_accura
cy: 0.7252
Epoch 20/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2673 - accuracy: 0.9689 - val_loss: 1.4663 - val_accura
cy: 0.7276
Epoch 21/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2631 - accuracy: 0.9695 - val_loss: 1.4389 - val_accura
cy: 0.7154
Epoch 22/25
5910/5910 [==============================] - 17s 3ms/sample - loss: 0.2700 - accuracy: 0.9663 - val_loss: 1.4041 - val_accura
cy: 0.7142
Epoch 23/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2696 - accuracy: 0.9673 - val_loss: 1.4802 - val_accura
cy: 0.7090
```

```
Epoch 24/25
5910/5910 [==============================] - 18s 3ms/sample - loss: 0.2469 - accuracy: 0.9750 - val_loss: 1.5829 - val_accura
cy: 0.7075
Epoch 25/25
5910/5910 [==============================] - 17s 3ms/sample - loss: 0.2422 - accuracy: 0.9773 - val_loss: 1.5228 - val_accura
cy: 0.7292
```

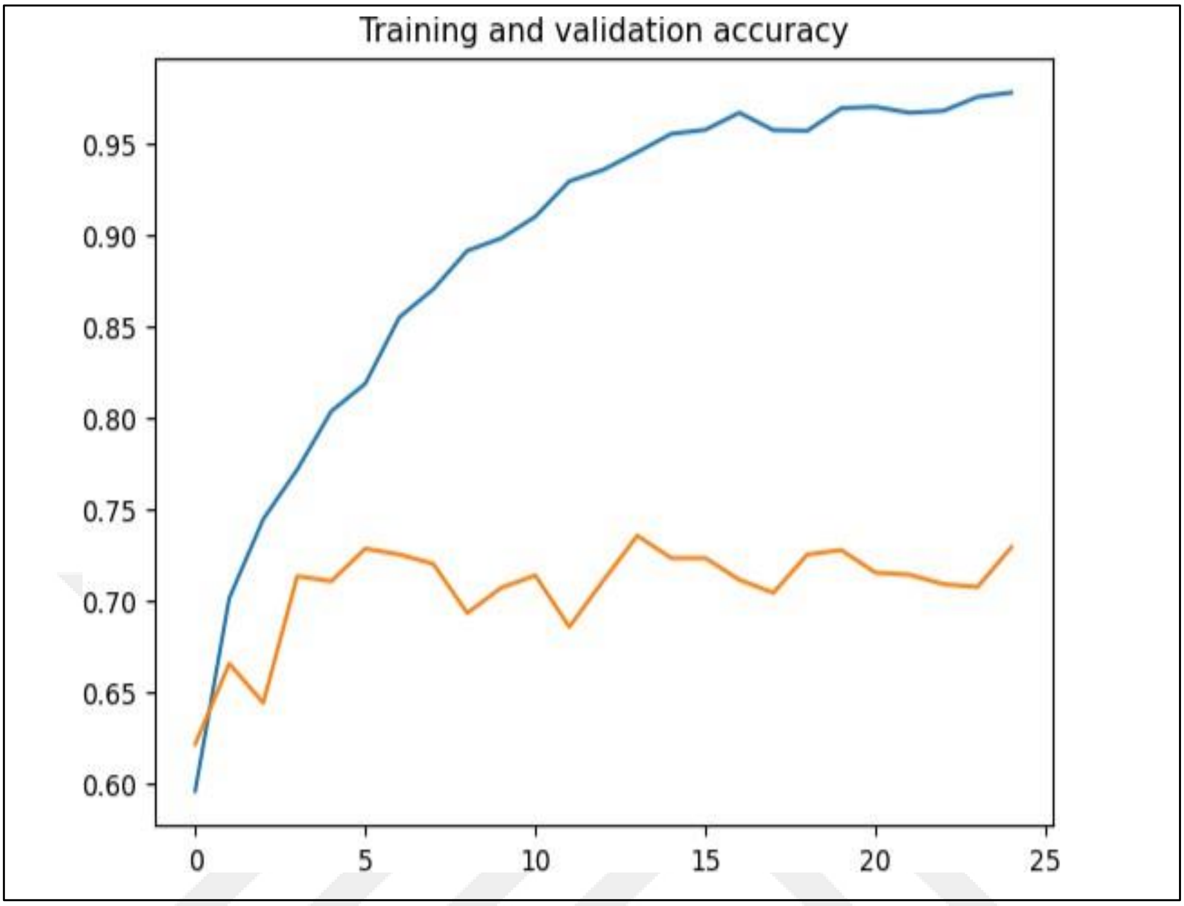

Figure 4.1: Accuracy Chart we Reached.

5. CONCLUSION AND FUTURE WORK

Using the results of the preprocessing, feature extraction, and classification, the recommended approach is concluded. The advantages of the learning algorithm include its infinite ability to make continuous, real-time diagnosis of heart arrhythmias, which helps patients who live in rural areas or in underdeveloped countries and cannot access heart disease care. The person using the circadian rhythm monitor will react immediately and alert emergency personnel to assist the person(s) in diagnosing an occupational arrhythmia when a potentially fatal heart rhythm manifests itself in high-risk groups. Through convolutional neural network layers, ECG devices have been developed, helping doctors in early detection of disease. We used samples from patients with heart disorders in the MIT-BIH database and inserted them into CNN layers. So, we made several changes to get the best result, getting 98% accuracy and 0.2% loss. As compared to other relevant works, our approach demonstrated the best performance and amazing results. Our upcoming research will test the suggested strategy using the 2D dataset.

REFERENCES

[1] 1.Jambukia, Shweta H., Vipul K. Dabhi, and Harshadkumar B. Prajapati. "Classification of ECG signals using machine learning techniques: A survey." *2015 International Conference on Advances in Computer Engineering and Applications*. IEEE, 2015.

[2] Peimankar, Abdolrahman, and Sadasivan Puthusserypady. "DENS-ECG: A deep learning approach for ECG signal delineation." *Expert systems with applications* 165 (2021): 113911.

[3] 3.Rajkumar, A., M. Ganesan, and R. Lavanya. "Arrhythmia classification on ECG using Deep Learning." *2019 5th international conference on advanced computing & communication systems (ICACCS)*. IEEE, 2019.

[4] Yamaç, Mehmet, et al. "A Personalized Zero-Shot ECG Arrhythmia Monitoring System: From Sparse Representation Based Domain Adaption to Energy Efficient Abnormal Beat Detection for Practical ECG Surveillance." *arXiv preprint arXiv:2207.07089* (2022).

[5] Bian, Yuexin, et al. "Identifying electrocardiogram abnormalities using a handcrafted-rule-enhanced neural network." *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2022).

[6] IMLE-Net: An Interpretable Multi-level Multi-channel Model for ECG Classification

[7] Mehari, Temesgen, and Nils Strodthoff. "Self-supervised representation learning from 12-lead ECG data." *Computers in biology and medicine* 141 (2022): 105114.

[8] Ma, Linhai, and Liang Liang. "Enhance CNN robustness against noises for classification of 12-lead ECG with variable length." *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2020.

[9] Wu, Yunan, et al. "A comparison of 1-D and 2-D deep convolutional neural networks in ECG classification." *arXiv preprint arXiv:1810.07088* (2018).

[10] Kachuee, Mohammad, Shayan Fazeli, and Majid Sarrafzadeh. "Ecg heartbeat classification: A deep transferable representation." *2018 IEEE international conference on healthcare informatics (ICHI)*. IEEE, 2018.

[11] Ullah, Amin, et al. "Classification of arrhythmia by using deep learning with 2-D ECG spectral image representation." *Remote Sensing* 12.10 (2020): 1685.

[12] Obeidat, Yusra, and Ali Mohammad Alqudah. "A Hybrid Lightweight 1D CNN-LSTM Architecture for Automated ECG Beat-Wise Classification." *Traitement du Signal* 38.5 (2021).

[13] Rubaiyat, Abu Hasnat Mohammad, et al. "Nearest subspace search in the signed cumulative distribution transform space for 1d signal classification." *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022.

[14] Nguyen, Thao, et al. "Detecting COVID-19 from digitized ECG printouts using 1D convolutional neural networks." *PLoS One* 17.11 (2022): e0277081.

[15] Butun, Ertan, et al. "1D-CADCapsNet: One dimensional deep capsule networks for coronary artery disease detection using ECG signals." *Physica Medica* 70 (2020): 39-48.

[16] Hasan, Md Abid, et al. "Cardiac Arrhythmia Detection in an ECG Beat Signal Using 1D Convolution Neural Network." *2020 IEEE Region 10 Symposium (TENSYMP)*. IEEE, 2020.

[17] Rabee, Ayman, and Imad Barhumi. "ECG signal classification using support vector machine based on wavelet multiresolution analysis." *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*. IEEE, 2012.

[18] Giannakakis, Giorgos, et al. "A novel multi-kernel 1D convolutional neural network for stress recognition from ECG." *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, 2019.

[19] Xiaolin, Li, Barry Cardiff, and Deepu John. "A 1d convolutional neural network for heartbeat classification from single lead ecg." *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2020.

[20] Le, Khiem H., et al. "Enhancing deep learning-based 3-lead ecg classification with heartbeat counting and demographic data integration." *arXiv preprint arXiv:2208.07088* (2022).

[21] Obeidat, Yusra, and Ali Mohammad Alqudah. "A Hybrid Lightweight 1D CNN-LSTM Architecture for Automated ECG Beat-Wise Classification." *Traitement du Signal* 38.5 (2021).

[22] Nguyen, Thao, et al. "Detecting COVID-19 from digitized ECG printouts using 1D convolutional neural networks." *PLoS One* 17.11 (2022): e0277081.

[23] Butun, Ertan, et al. "1D-CADCapsNet: One dimensional deep capsule networks for coronary artery disease detection using ECG signals." *Physica Medica* 70 (2020): 39-48.

[24] Butun, Ertan, et al. "1D-CADCapsNet: One dimensional deep capsule networks for coronary artery disease detection using ECG signals." *Physica Medica* 70 (2020): 39-48.

[25] Saini, Indu, Dilbag Singh, and Arun Khosla. "QRS detection using K-Nearest Neighbor algorithm (KNN) and evaluation on standard ECG databases." *Journal of advanced research* 4.4 (2013): 331-344.

[26] Venkatesan, C., et al. "ECG signal preprocessing and SVM classifier-based abnormality detection in remote healthcare applications." *IEEE Access* 6 (2018): 9767-9773.

[27] Naik, G. Rajender, and K. Ashoka Reddy. "Comparative analysis of ECG classification using neuro-fuzzy algorithm and multimodal decision learning algorithm: ECG classification algorithm." *2016 3rd International Conference on Soft Computing & Machine Intelligence (ISCMI)*. IEEE, 2016.

[28] Chollet, Francois. *Deep learning with Python*. Simon and Schuster, 2021.

[29] Chollet, F. (2021). *Deep learning with Python*. Simon and Schuster.

[30] Brownlee, J. (2016). *Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras*. Machine Learning Mastery.

[31] Essa, Ehab, and Xianghua Xie. "An ensemble of deep learning-based multi-model for ECG heartbeats arrhythmia classification." *IEEE Access* 9 (2021): 103452-103464.

[32] Rai, Hari Mohan, Kalyan Chatterjee, and Chandra Mukherjee. "Hybrid CNN-LSTM model for automatic prediction of cardiac arrhythmias from ECG big data." *2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. IEEE, 2020.

[33] Xu, Xuexiang, and Hongxing Liu. "ECG heartbeat classification using convolutional neural networks." *IEEE Access* 8 (2020): 8614-8619.

[34] Denysyuk, Hanna Vitaliyivna, et al. "Algorithms for automated diagnosis of cardiovascular diseases based on ECG data: A comprehensive systematic review." *Heliyon* (2023).

[35] Ahmed, Adel A., et al. "Classifying Cardiac Arrhythmia from ECG Signal Using 1D CNN Deep Learning Model." *Mathematics* 11.3 (2023): 562.

[36] Zhao, Yunxiang, et al. "ECG classification using deep CNN improved by wavelet transform." *Computers, Materials and Continua* (2020).

[37] Takalo-Mattila, Janne, Jussi Kiljander, and Juha-Pekka Soininen. "Inter-patient ECG classification using deep convolutional neural networks." *2018 21st Euromicro Conference on Digital System Design (DSD)*. IEEE, 2018.

[38] Ebrahimi, Z., Loni, M., Daneshtalab, M., & Gharehbaghi, A. (2020). A review on deep learning methods for ECG arrhythmia classification. *Expert Systems with Applications: X*, *7*, 100033.

[39] Murat, Fatma, et al. "Application of deep learning techniques for heartbeats detection using ECG signals-analysis and review." *Computers in biology and medicine* 120 (2020): 103726.

[40] Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. "A novel application of deep learning for single-lead ECG classification." *Computers in biology and medicine* 99 (2018): 53-62.

[41] Murat, Fatma, et al. "Application of deep learning techniques for heartbeats detection using ECG signals-analysis and review." *Computers in biology and medicine* 120 (2020): 103726.

[42] Ebrahimi, Zahra, et al. "A review on deep learning methods for ECG arrhythmia classification." *Expert Systems with Applications: X* 7 (2020): 100033.

[43] Sannino, Giovanna, and Giuseppe De Pietro. "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection." *Future Generation Computer Systems* 86 (2018): 446-455.

[44] Al Rahhal, Mohamad Mahmoud, et al. "Deep learning approach for active classification of electrocardiogram signals." *Information Sciences* 345 (2016): 340-354.

[45] Wieclaw, Lukasz, et al. "Biometrie identification from raw ECG signal using deep learning techniques." *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. Vol. 1. IEEE, 2017.

[46] Śmigiel, Sandra, Krzysztof Pałczyński, and Damian Ledziński. "Deep learning techniques in the classification of ECG signals using r-peak detection based on the PTB- XL dataset." *Sensors* 21.24 (2021): 8174.

[47] Swapna, G., K. P. Soman, and R. Vinayakumar. "Diabetes detection using ecg signals: An overview." *Deep Learning Techniques for Biomedical and Health Informatics* (2020): 299-327.