TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PRUWANCHAL CAMPUS, DHARAN-8

A

FINAL YEAR PROJECT REPORT

ON

REAL-TIME ARRHYTHMIA DETECTION USING CNN ON ECG SIGNALS

SUBMITTED BY:

DIPESH CHANDRA JHA (075BEI010)

PRASOON JHA (075BEI027)

RAJIV KUSHWAHA (075BEI029)

SATRODHAN KUMAR YADAV (075BEI036)

SUBMITTED TO:

THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULLFILLMENT OF THE REQUIREMENT FOR THE BACHELOR DEGREE OF ELECTRONICS COMMUNICATION AND INFORMATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

IOE-PURWANCHAL CAMPUS, DHARAN-8

May 23, 2023

DEPARTMENTAL ACCEPTANCE

The report entitled **"REAL-TIME ARRHYTHMIA DETECTION USING CNN ON ECG SIGNALS"**, submitted by Prasoon Jha, Rajiv Kushwaha, Dipesh Chandra Jha and Satrodhan Kumar Yadav in the partial fulfilment for the award of the degree of "Bachelor in Electronics, communication and Information Engineering" has been accepted as a bona fide record of work independently out by them in the department.

Manoj Kumar Guragai

Er. Manoj Kumar Guragai

Assistant Professor

Head of the Department

Department of Electronics and Computer Engineering

Purwanchal Campus, Dharan-Nepal

TRIBHUWAN UNIVERSITY

INSTITUTE OF ENGINEERING

PRUWANCHAL CAMPUS, DHARAN-8

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

STUDENT DECLARATION

We hereby declare that we are the only author of this complete work and that no sources other than those listed here have been used in this work.

Name of the Students:

Dipesh Chandra Jha (40858)

Prasoon Jha (40875)

Rajiv Kushwaha (40877)

Satrodhan Kumar Yadav (40884)

23 MAY, 2023

LETTER OF APPROVAL

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PURWANCHAL CAMPUS, DHARAN-8

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certify that they have read, and recommended to the Institute of Engineering for acceptance, a project entitled "**REAL-TIME ARRHYTHMIA DETECTION USING CNN ON ECG SIGNALS"** submitted by Dipesh Chandra jha, Prasoon Jha, Rajiv Kushwaha, and Satrodhan Kumar Yadav in partial fulfilment of the requirements for the Bachelor's Degree in Electronics, Communication and Information Engineering.

Prof. Ram Krishna Maharjan

External Examiner, Reader

Department of Electronics, Communication and Information Engineering, Pulchowk Campus, IOE, Dharan

Er. Om Prakash Dhakal

Supervisor, Assistant Professor

Department of Electronics and Computer Engineering, Purwanchal Campus, IOE, Dharan

Er. Manoj Kumar Guragai

Assistant Professor, Head of Department

Department of Electronics and Computer Engineering, Purwanchal Campus, IOE, Dharan

©COPYRIGHT

Head of Department

Department of Electronics and Computer Engineering

Purwanchal Campus, Institute of Engineering

Dharan, Sunsari

Nepal

ACKNOWLEDGEMENT

We owe our deepest gratitude to the Department of Electronics and Computer Engineering, IOE, Purwanchal Campus for providing us with an opportunity to work on a major project as part of our syllabus. We are heartily indebted to our project supervisor Mr. Om Prakash Dhakal for his constant support and guidance throughout this project. It was his valuable suggestions that helped us to cope up with the emerging obstacles during the development of this project.

We are grateful to our Head of Department Mr. Manoj Kumar Guragai for his interest in our project that motivated us to put more effort into this project. Besides the project tenure provided by the department has great support for the completion of our project. We are also grateful to all our teachers for their suggestions and inspirational lectures that paved the way towards the completion of this project.

We express our gratitude towards our External Examiner Mr. Ram Krishna Maharjan and our Project Coordinator for their valuable suggestions.

We would like to convey our thanks to the teaching and non-teaching staff of the Department of Electronics and Computer Engineering, Purwanchal Campus for their valuable help and support throughout our project.

We would like to express our deepest appreciation to every one of those helping hands who helped us in every step of our project.

ABSTRACT

Electrocardiogram (ECG) arrhythmia classification is a critical task in clinical cardiology. Traditional methods for arrhythmia classification require extensive domain knowledge and manual feature extraction, which can be time-consuming and prone to errors. Although, a lot of research and studies are designed for observing the ECG signal remotely, there are very less proposed methods for classifying these signals with monitoring, and therefore, to design complete health care system, classification techniques should be used to classify the extracted signal. In recent years, deep learning-based approaches, such as convolutional neural networks (CNNs), have shown promising results in ECG arrhythmia classification. In this project, we used AD8232 sensor to extract the ECG signal, ESP32 to send the extracted ECG signal to our own website, in the website we used Band pass filter to filter the signal, then BioSPPY function (detects R-peak and create 3-seconds window to left and 3-seconds window to right) and create input test signal for our model, to classify the test signal we use 1-D CNN architecture. This model is designed to automatically learn discriminative features from raw ECG signals without manual feature extraction. We evaluated performance of the model trained on the publicly available MIT-BIH Arrhythmia dataset and achieved an overall 75% of test accuracy. Our result demonstrated that our model can effectively classify ECG arrhythmia and has the potential to improve the efficiency and accuracy of clinical diagnosis. This model could be a valuable tool for cardiologists in identifying arrhythmias and providing appropriate treatment. The algorithm is implemented and performed by using python.

Keywords: ECG classifications, 1-D CNN, MIT-BIH dataset, ESP32, AD8232 heart sensor, arrhythmia, BioSPPY functions.

,

TABLE OF CONTENTS

DEPARTMENTAL ACCEPTANCE i

LETTER OF APPROVAL iii

©COPYRIGHT........ iv

ACKNOWLEDGEMENT v

ABSTRACT........ vi

TABLE OF CONTENTS........ vii

LIST OF FIGURES ix

ABBREVIATIONS x

1. INTRODUCTION 1

1.1 BACKGROUND 1

1.2 PROBLEM STATEMENT 1

1.3 OBJECTIVES 2

1.4 APPLICATION 2

1.5 FEATURES 3

1.6 FEASIIBILITY ANALYSIS 4

1.6.1 ECONOMIC FEASIBILITY 4

1.6.2 TECHINICAL FEASIBILITY 4

1.6.3 OPERATIONAL FEASIBILITY 4

1.7 SYSTEM REQUIREMENTS 5

1.7.1 SOFTWARE REQUIREMENTS 5

2. LITERATURE REVIEW 9

3. METHODOLOGY 11

3.1 WORKING PRINCIPLE 11

3.2 DATASET COLLECTION AND PREPARATION 13

3.3 FLOWCHART AND ALGORITHM 14

3.4 SYSTEM DESIGN AND IMPLEMENTATION 15
3.5 SOFTWARE DEVELOPMENT MODEL 17
3.6 MODEL ARCHITECTURE 18
4. RESULT AND DISCUSSION 19
4.1 OUTPUT 19
4.2 CLASSIFICATION REPORT 20
4.3 TRAINING LOSS CURVE 21
4.4 CONFUSION MATRIX 21
4.5 ROC CURVE 22
4.6 LIMITATIONS 22
4.7 PROBLEMS FACED 23
4.8 BUDGET ANALYSIS 23
5 CONCLUSION AND FUTURE ENHANCEMENT 24
5.1 CONCLUSION 24
5.2 FUTURE ENHANCEMENT 24
References 25

LIST OF FIGURES

Figure 1 : ESP32	7
Figure 2 : AD8232 sensor	8
Figure 3 : ECG categorization	13
Figure 4 : Flowchart part of the system	14
Figure 5 : Overview of our system	15
Figure 6 : Overview of Agile model	17
Figure 7 : CNN model architecture	18
Figure 8 : Normal heart condition	19
Figure 9 : Abnormal heart condition	19
Figure 10 : Training loss curve	21
Figure 11 : Confusion matrix	21
Figure 12 : ROC curve	22

ABBREVIATIONS

ARR	Cardiac Arrhythmia
BioSPPY	Bio signal Processing in Python
CANFIS	Co-Active Neuro-Fuzzy Inference System
CNN	Convolutional Neural Networks
CSS	Cascading Style Sheets
CVD	Cardiovascular Disease
DNN	Deep Neural Network
DT	Decision Tree
ECG	Electrocardiogram
FP	False Positive
FN	False Negative
HTML	Hypertext Markup Language
IOT	Internet of Things
KNN	K-Nearest Neighbors
MIT-BIH	Massachusetts Institute of Technology-Beth Israel Hospital
NSR	Normal Sinus Rhythm
RF	Random Forest
SVM	Support Vector Machine
TP	True Positive
TN	True Negative
VS code	Visual Studio code
Wi-Fi	Wireless Fidelity

1. INTRODUCTION

1.1 BACKGROUND

The body tissues and fluid conduct electricity well show the electrical activity in the heart can be recorded on the skin surface using electrodes position on the limbs or chest. This graphical representation of electrical activity of the heart is called Electrocardiogram (ECG/EKG). Nowadays, people may take their blood pressure in the comfort of their house however, ECG is more crucial than blood pressure for real time monitoring of heart function. An ECG records these impulses to show how fast the heart is beating, the rhythm of the heart beats and the strength and timing of the electrical impulses as they move through the different part of the heart. This is required when there is chest pain, heart palpitations etc. It helps doctors to determine how certain heart disease is.

Hence, this system will be developed using CNN algorithm where ECG signal from IOT device will be fed to the algorithm to determine regularity and irregularity of the heart rhythm.

1.2 PROBLEM STATEMENT

Before ECG was discovered it was really difficult to know everything about the condition of heartbeat. Only experienced doctors could figure out through the patient's symptoms that they had previously encountered for different cases of different patients and used to prescribe medicine according to that. As time passes, the number of heart patients is increasing rapidly due to which there will be queue of people to have ECG in the hospital which will be time consuming. Also, heart patients can have their ECG at home at regular intervals of time. Our model overcomes this problem by taking ECG ourselves and classifying whether it is normal or abnormal. So, patients only must visit doctor when the ECG is abnormal which reduces the frequency of visiting doctor, doctor's time, patient's time, patient's money etc.

1.3 OBJECTIVES

- To classify normal and abnormal heart rhythm conditions using different CNN architectures.
- To make ECG testing handy, user friendly, economical and easy for user to understand and to understand and interpret his /her condition.

1.4 APPLICATION

- Automated arrhythmia classification: The proposed model can automate the process of arrhythmia classification, reducing the need for manual analysis by cardiologists. This can save time and improve the accuracy of diagnosis.
- Early detection of arrhythmias: The proposed model can detect arrhythmias in ECG signals at an early stage, enabling timely diagnosis and treatment. This can improve patient outcomes and reduce the risk of complications.
- Personalized treatment: The proposed model can classify ECG arrhythmias based on their type and severity, enabling personalized treatment for individual patients.
- Telemedicine: The proposed model can be integrated into telemedicine platforms, enabling remote monitoring of ECG signals and real-time arrhythmia classification. This can be particularly beneficial for patients in remote areas or those with limited access to healthcare facilities.

1.5 FEATURES

This model for ECG arrhythmia classification is designed to automatically learn discriminative features from preprocessed ECG signals. Some of the features of this model are:

- IOT sensors: ECG sensors are placed on the patient's chest to capture the ECG signals, which are transmitted to a central server or cloud-based system through wireless or wired networks.
- Data pre-processing: The ECG signals are pre-processed to remove any noise or artifacts that may affect the accuracy of the classification.
- CNN-based classification: A CNN is trained using a large dataset of ECG signals to classify the signals into various categories such as normal sinus rhythm, atrial fibrillation, or ventricular tachycardia.
- Real-time monitoring: The IOT-based ECG monitoring system enables continuous monitoring of patients in real-time, with alerts generated in case of any abnormality in the ECG signals.
- Remote access: The ECG signals can be accessed remotely by healthcare professionals, enabling them to monitor patients from anywhere, anytime.

1.6 FEASIIBILITY ANALYSIS

1.6.1 ECONOMIC FEASIBILITY

The cost of the software, hardware, and people needed for the suggested model for ECG arrhythmia classification relies on the economic viability of its development and implementation. The suggested approach is anticipated to be less expensive than conventional techniques that need manual feature extraction and in-depth domain expertise, despite the fact that hardware and software costs might be substantial. The cost of hardware may be reduced by employing already-existing computing resources, and the model can be trained on desktop PCs or cloud-based servers.

1.6.2 TECHINICAL FEASIBILITY

The availability of preprocessed ECG data and the creation of a reliable and effective CNN model architecture are prerequisites for the proposed model's technical viability. Due to noise and artifacts, filtering ECG data may be difficult, however there are a number of preprocessing techniques available to overcome these difficulties as ECG signal is highly sensitive to Physical movement, Electrical interferences, and other sorts of noise. Numerous open-source tools and frameworks are available that help streamline the construction process, but deep learning and signal processing knowledge are required to create a reliable and effective CNN model.

1.6.3 OPERATIONAL FEASIBILITY

In this project, the heart rate is measured using an AD8232 sensor. As the sensor employed here doesn't provide a noise-free signal when used for medical purposes, it is only being used for research purposes. Despite the presence of noise, we can accurately categorize normal and abnormal conditions using the 1-D CNN pre-trained model. To train the model, we use the MIT-BIH Arrhythmia dataset. This project provides a convenient, inexpensive way to measure heart rate with an estimated 75% test accuracy.

1.7 SYSTEM REQUIREMENTS

1.7.1 SOFTWARE REQUIREMENTS

1.7.1.1 GOOGLE COLAB

Google Colaboratory, also known as Google Colab, is a free online platform that provides users with a Jupyter notebook environment in which they can run Python code. It is designed to support machine learning, data analysis, and other scientific computing applications. It helps to train the machine learning model with higher degree of feasibility. It allows users to write and execute Python code in a browser, eliminating the need for local installations or configurations.

1.7.1.2 VS CODE

Visual Studio Code (VS code) is a popular and powerful code editor that provides a lightweight, fast, and customizable development environment for a wide range of programming languages. It includes features such as IntelliSense, built-in debugging, source control integration, extension marketplace, customizability, and cross-platform support. It is also used for web development. It has a number of features and extensions that make it well-suited for developing web applications, including support for popular web technologies such as HTML, CSS, and JavaScript.

1.7.1.3 PROGRAMMING TOOLS

- **PYTHON**

  Python is a high-level, interpreted programming language that is widely used in data science, machine learning, web development, scientific computing, and other areas of software development. Python has a large standard library that provides many useful tools and modules for a wide range of programming tasks. Python language is used to train our model especially backend through flask. Python has a rich ecosystem of third-party libraries and frameworks that extend its capabilities in various domains. It is cross-platform compatible and has a strong and supportive community.

- **FLASK**

  Flask is used as a backend web framework in Python for developing web applications. Flask is designed to be lightweight and flexible, making it well-suited for developing small to medium-sized web applications that require a backend server. It also provides built-in support for routing, request handling, and response generation, making it easy to create a RESTful API for the frontend of a web application. The Flask community is vibrant and active, with extensive documentation, tutorials, and a wide range of third-party extensions available. Flask allows developers to create web applications and APIs quickly and efficiently. Flask allows developers to map URLs to specific Python functions called view functions. These view functions can generate dynamic content or render templates to produce HTML responses. Flask also supports URL parameter capturing and variable rules for more flexible routing.

- **HTML/CSS/JAVASCRIPT**

  HTML is used to create web pages and web applications. It provides a way to structure content on a web page, making it easier to read and understand. HTML has a simple and intuitive syntax that makes it easy for beginners to learn and understand. Cascading Style Sheets is a styling language used to describe the presentation of a document written in HTML. JavaScript is a client-side scripting language, which means that it is executed by the web browser on the client's computer rather than on the server. This allows for the creation of dynamic and interactive user interfaces, such as drop-down menus, pop-ups, and sliders that respond to user input without requiring a page reload.

1.7.2 HARDWARE REQUIREMENTS

1.7.2.1 ESP32

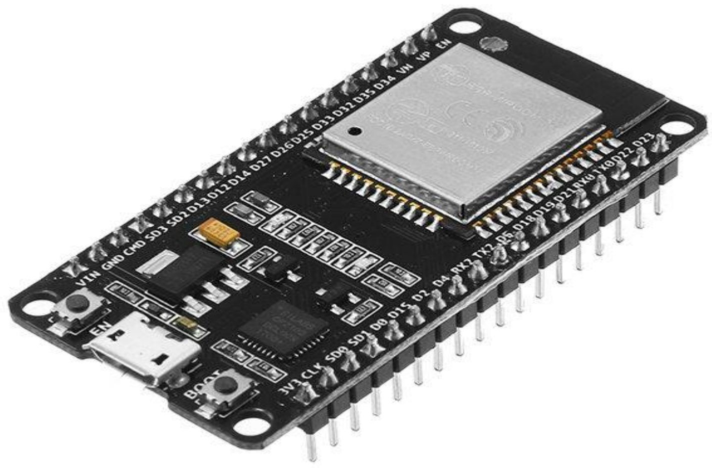

Figure 1 :ESP32

ESP32 is a series of low-cost, low-power systems on a chip microcontroller with integrated Wi-Fi and dual-mode Bluetooth. It achieves very low power consumption through power saving features including clock synchronization and multiple modes of operation. The ESP32 chip's quiescent current is less than 5 μA which makes it the ideal tool for your battery powered projects or IoT applications. It is used to send the ECG signal / analog value from AD8232 sensor to our website with its Wi-Fi inbuilt feature. It has built-in Wi-Fi and Bluetooth capabilities, making it suitable for Internet of Things (IoT) applications. The module also provides a range of peripherals and interfaces, such as GPIOs, UART, I2C, SPI, ADC, DAC, and more. The ESP32 can be programmed using various development frameworks and programming languages, including the Arduino IDE, ESP-IDF (Espressif IoT Development Framework), MicroPython, and Lua.

1.7.2.2 AD8232 SENSOR

Figure 2: AD8232 sensor

A board that is used to gauge the electrical activity of the heart is the AD8232 Heart Rate Monitor. An ECG or electrocardiogram can be used to track this electrical activity and produce an analog reading. The AD8232 single lead heart rate monitor functions as an op amp to assist in obtaining a clear signal from the PR and QT Intervals with ease. ECGs can be very noisy. An integrated signal conditioning block called the AD8232 is used in applications for bio potential measurements like ECGs. Disposable ECG electrodes are used as a connector with snap-connect lead wires to this sensor. The AD8232 sensor can interface with microcontrollers, development boards, and other electronic devices through standard analog interfaces like SPI (Serial Peripheral Interface) or I2C (Inter-Integrated Circuit). This allows for easy integration into various projects and systems.

2. LITERATURE REVIEW

The article in [1] provides a strategy for categorizing ECG data based on autoregressive (AR) analysis and signal characteristics extraction. To enhance the interpretation of ECG data, the suggested method employs two-event-related moving averages (TERMAs) and fractional Fourier transform (FFT) methods. The study demonstrates effective categorization using the derived AR parameters and emphasizes the benefits of the suggested strategy over conventional techniques.

[2] Focuses on enhancing patient health monitoring systems by integrating an AD8232 ECG Sensor with a NodeMCU ESP8266 Board. It shows how to interface the ECG waveform with the board and display it on a Serial Plotter Screen. It also emphasizes the potential to send the ECG waveform to an IoT cloud platform, allowing remote monitoring of the signal from any place using a PC or smartphone. This development eliminates the need for patients to stay in the hospital for continuous cardiac activity monitoring, providing them with greater convenience and access to healthcare services.

The [3] paper proposes a 12-layer deep one-dimensional convolutional neural network (CNN) for the classification of five micro-classes of heartbeat types in the MIT-BIH Arrhythmia database. In the experiments, the paper used a wavelet self-adaptive threshold denoising technique. The suggested model outperforms the BP neural network, random forest, and other CNN networks in terms of accuracy, sensitivity, robustness, and anti-noise capabilities, according to comparative research. The model's ability to accurately classify data shows that it has the potential to reduce medical resource use and enhance clinical practice. The model's efficiency in categorizing different types of heartbeats is further supported by its high positive rate, specificity, and overall classification accuracy rate of 97.41%.

[4] Focuses on applying a 1-D convolutional neural network to classify arrhythmias in electrocardiogram (ECG) data. The study makes use of many PhysioNet datasets, particularly LEAD1 and LEAD2, as input signals. Further insights into the categorization process are provided by the paper's use of symbolic regression and correlation analysis to analyze the outcomes and behavior of the neural network. This

study makes a contribution to our understanding and interpretation of deep learning-based ECG categorization.

[5] Discusses 1-dimensional convolutional neural network (CNN) method to categorize heartbeats in ECG data acquired from ambulatory devices. Data augmentation using the SMOTE technique is used to resolve the dataset's class imbalance. With an accuracy of 98.12%, sensitivity of 98.07%, and specificity of 98.29%, the trained network performs well.

This article [6] describes a technique that categorizes different ECG image patterns for anesthesia evaluation using convolutional neural networks (CNNs). The project focuses on creating prototypes for Internet of Things (IoT)-based ECG signal monitoring systems. The study uses deep neural networks to categorize several signal patterns, such as QRS broadening, sinus rhythm, ST depression, and ST elevation. Utilizing half of the training and test sets, three models—ResNet, AlexNet, and SqueezeNet—are created and assessed. It is reported on the models' precision and kappa statistics for classifying ECG waveforms. The study shows the viability of IoT-based real-time ECG monitoring and the potential of deep neural network models for categorizing ECG picture kinds

3. METHODOLOGY

3.1 WORKING PRINCIPLE

The hardware and software components will be combined to complete the ECG monitoring system. Transmission of data will occur through the internet. The Hardware part will be divided into 2 major components. The first one is AD8232 ECG sensor also known as heart rate monitor chip which is responsible for recording cardiac electrical activity of the patient. Every heartbeat is recorded using an ECG sensor with disposable electrodes that are attached to the chest. Heart rhythm will be converted to an electrical signal by the ECG sensor's electrodes. ECG sensors are incredibly thin, light, and accurate at measuring continuous heartbeat and providing heartbeat rate data. Second is ESP32 which transfers electrical signal (ECG waveform) to external computer which will allow users to see the data to be analyzed remotely.

The ESP32 microcontroller and AD8232 sensor collected the ECG data, which will be sent into the AI module, which uses CNN for classification analysis, to automate the analysis process. Four stages make up the CNN model's ECG analysis: ECG will be done in four stages:

1. **ECG pre-processing:** This stage increases the overall quality of ECG signals for better analysis and examination. Filtering is the most popular way for removing excessive noise from ECG readings like use of various filtering and denoising technique like Wavelet denoising. Preprocessing is done in the further steps too where we use an algorithm for extraction and optimization.

2. **Detection of QRS:** QRS complex was the single greatest critical element of ECG signal. For a remote electrocardiogram (ECG) monitoring application, QRS detection is a preliminary step for detecting the heartbeat for the subsequent rhythm classification, so a high QRS detection rate method is the most significant part of the ECG analysis algorithm. So, for that WFDB library is used which implements Pan-Tompkins algorithm to obtain symbols from ECG signal of dataset we used.

- **WFDB Library:**

  The WFDB library offers tools for reading, writing, and processing physiological information, including ECG signals. It contains tools for reading ECG data from a variety of file formats, including the PhysioBank format and the MIT-BIH Arrhythmia Database format, as well as performing fundamental signal processing tasks including filtering and segmentation. For working with physiological data, the research field makes significant use of the WFDB library. It helps in reading the annotation file of each ECG recording and extract the beat annotations using the 'atr' annotation type. The beat annotations are then processed further to identify normal beats, abnormal beats, and non-beats.

- **Pan-Tompkins algorithm:**

  The Pan-Tompkins method was created particularly for detecting QRS complexes in ECG data. It focuses on recognizing the QRS complexes, which indicate the ventricular depolarization in the heart. Band pass filtering, differentiation, squaring, integration, and thresholding are all phases in the process. The Pan-Tompkins algorithm's major purpose is to reliably detect the R-peaks (the highest point of the QRS complex) in the ECG signal. The method can calculate the heart rate and extract information linked to the QRS complex, such as duration and amplitude, by recognizing the R-peaks.

3. **Extraction of the parameters:** For the ECG signal to be categorized correctly, the significant parameters must be extracted like beginning peak and end of QRS complex, RR intervals and other significant features required for better result in classification.

4. **Extracted parameter classification and clustering**: Various approaches can be used to categorize and identify the gathered ECG signal features like in our project we used 1D CNN to classify the ECG signal as normal or abnormal since the ECG sensor we used is single lead and capable of only binary categorization.

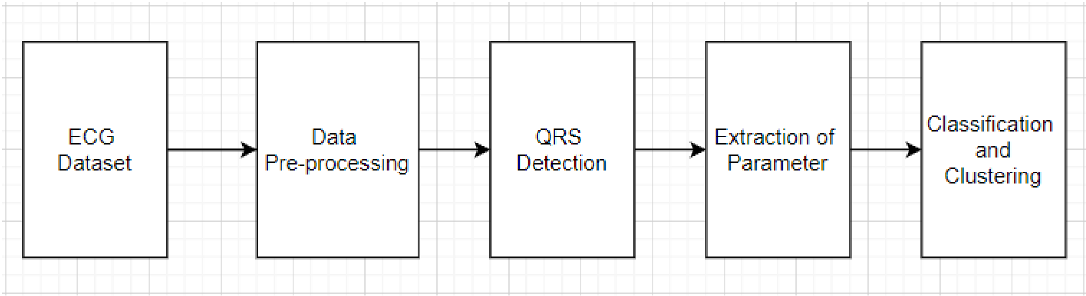

Figure 3 : ECG categorization

3.2 DATASET COLLECTION AND PREPARATION

We have used MIT-BIH arrhythmia dataset (48 recordings) and recordings were digitized at 360 samples per second of 30 minutes. We have used this to classify ECG signals into two categories.

- NSR
- ARR

We have implemented an architecture of 1-D CNN model with split between trained, test and valid dataset on 48 recordings to prevent data leakage. Segmentation is performed on these recordings creating approximately 100000 heartbeats distributed in 16 different classes. But we have pre-processed the signal and categorized it into Normal beat, Abnormal beat and non-beat in which non-beats are ignored. We also performed data augmentation on training samples to increase the number of samples for training the model. We also use Gaussian mixture in our training dataset and then normalized the signal in the range of [-1, +1].

3.3 FLOWCHART AND ALGORITHM

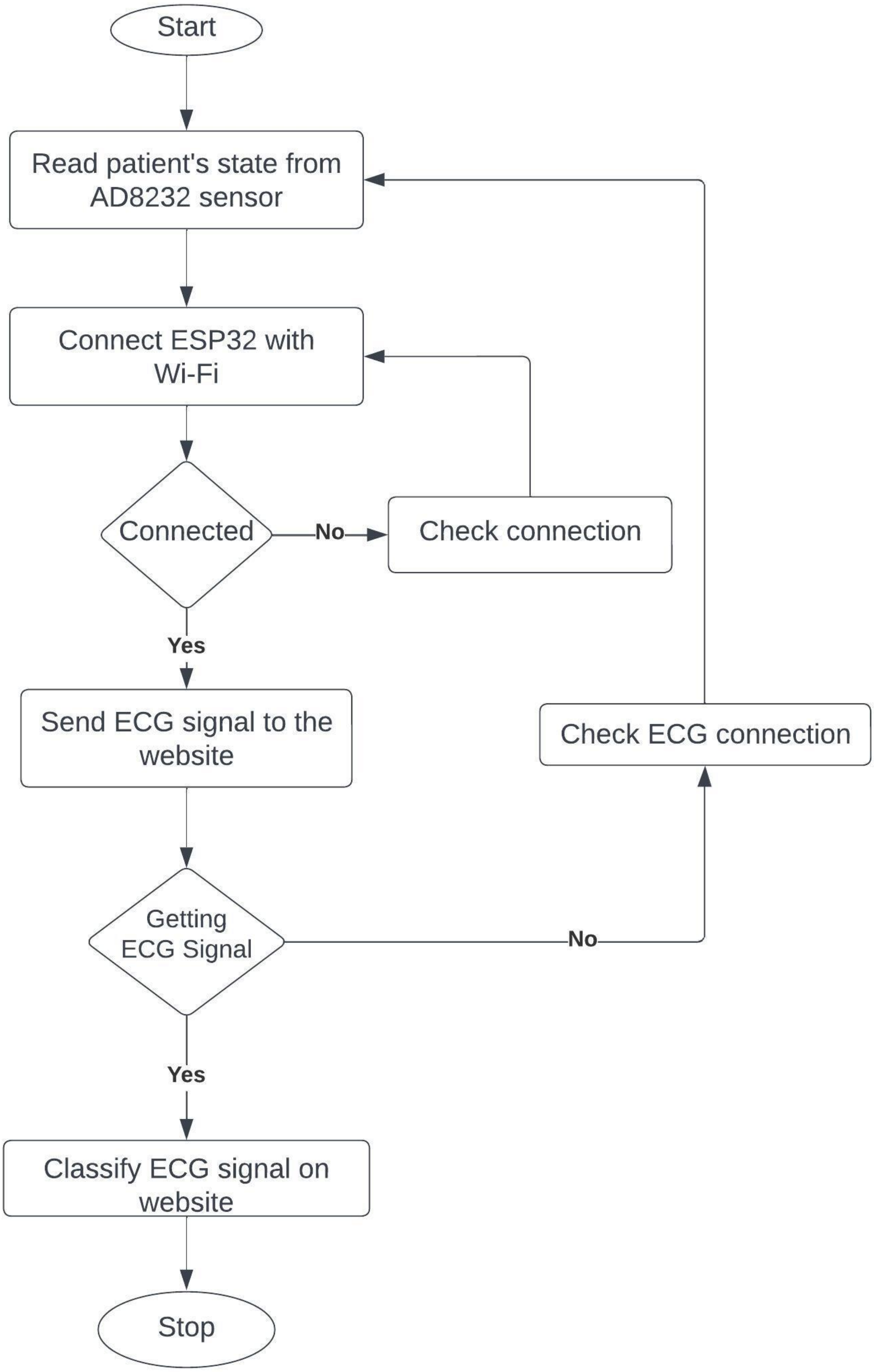

Figure 4 : Flowchart part of the system

First, The ECG data is collected in real time by connecting ESP32 and AD8232 sensors to the human body. We encode that signal into base64 string that is suitable for our website.

We have used Flask/python for our backend that receives encoded base64 string from the website and our server will decode it and feed it to pre-processing and classification function to predict the label and provide the response to the front end created using HTML, CSS and JavaScript to display the classification output on web based system using the graphical user interface.

3.4 SYSTEM DESIGN AND IMPLEMENTATION

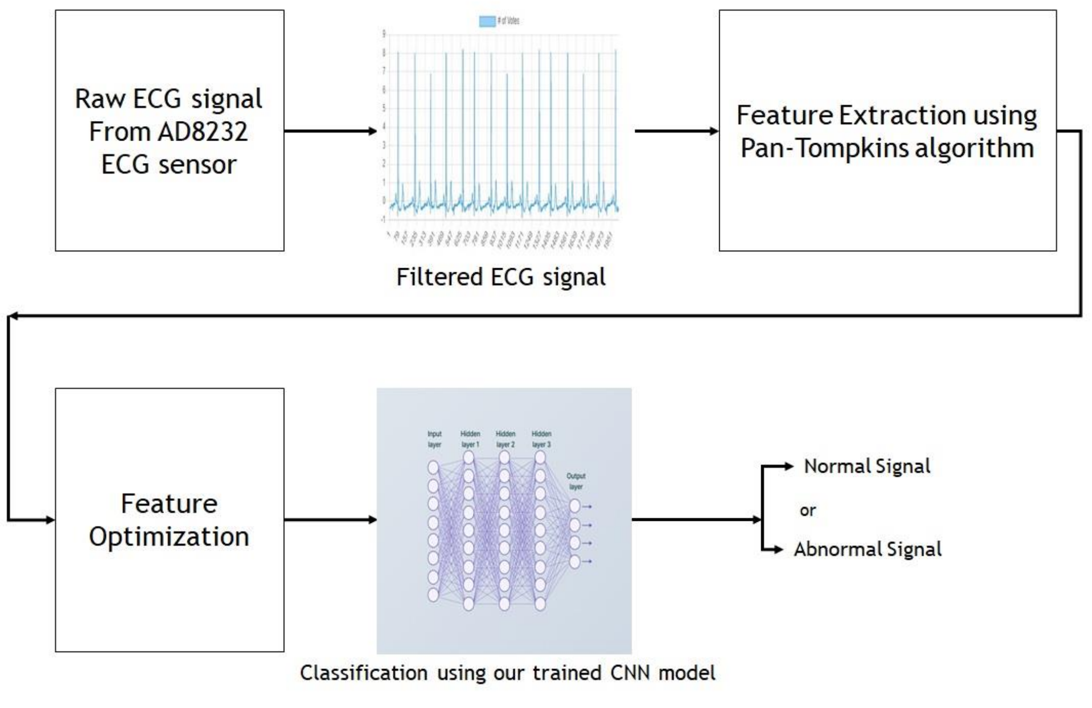

Figure 5: Overview of our system

The raw ECG data is captured by our system's AD8232 ECG sensor and then readily delivered over the internet using an ESP32 microcontroller. The raw signal is subjected to important preprocessing procedures, such as several filtering techniques, the Fourier transform, the Hilbert transform, and Denoising procedures as well for efficient noise reduction and artifact removal, resulting in a filtered signal.

After that, the well-known Pan-Tompkins method is used to extract relevant features from the filtered ECG data, such as QRS complex identification. A complete feature optimization technique is used to improve these retrieved features, improving the signal representation for later analysis. Notably, for best classification results, numerous essential parameters such as amplitude, R-R interval, P wave duration, QRS complex duration, discrepancies in the ST segment such as elevation, and others are taken into considerations.

Finally, the system includes a 1D convolutional neural network (CNN) model that has been trained on the well-known MIT-BIH dataset. This model, which is implemented on a Flask backend, allows the categorization of ECG data into two categories: normal and abnormal. The trained CNN model has great accuracy and strength, allowing for real-time ECG data classification with little delay.

The block diagram above depicts a complete framework for developing a remote ECG monitoring system capable of accurate real-time categorization by combining hardware, signal processing, and machine learning algorithms.

3.5 SOFTWARE DEVELOPMENT MODEL

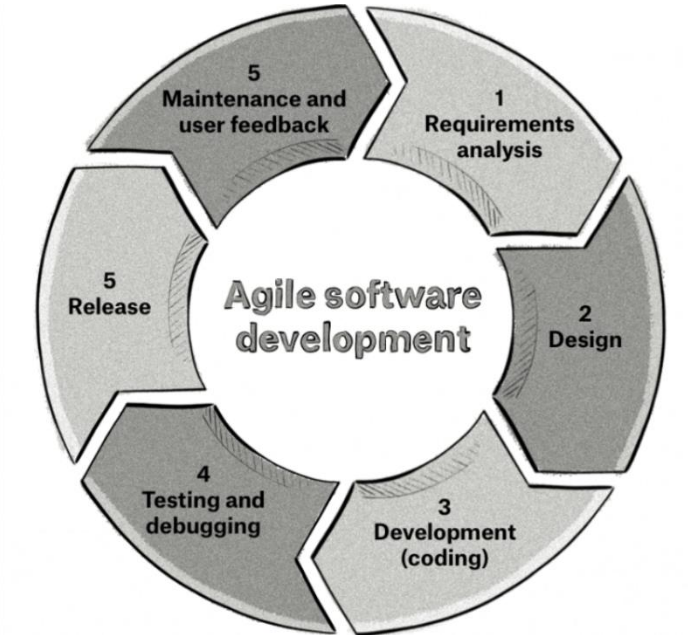

Figure 6: Overview of Agile model

Agile software development is a flexible and iterative method that focuses on producing working software in brief phases. I used the Agile technique to efficiently construct and deliver my website in the context of classifying real-time ECG signal.

The Agile development approach divides the development process into smaller increments, allowing for constant input and adaptability. This method allowed us to adjust fast to changing needs and prioritize features based on their value and effect. Our team collaborated throughout the development process to establish and enhance the website's functionality, ensuring it met the required objectives. We were able to break down the website development into manageable tasks and prioritize them based on their importance by utilizing Agile. This iterative process enabled regular testing and validation, ensuring that the website's capabilities were effectively constructed and integrated.

A high-quality finished product that is in compliance with the project's objectives and specifications is produced as a consequence of this iterative development process, which encourages flexibility and adaptability.

3.6 MODEL ARCHITECTURE

Model: "model_1"

Layer (type)	Output Shape	Param #
inputs_cnn (InputLayer)	[(None, 2160, 1)]	0
conv1d_7 (Conv1D)	(None, 2155, 64)	448
batch_normalization_3 (BatchNormalization)	(None, 2155, 64)	256
max_pooling1d_3 (MaxPooling1D)	(None, 1078, 64)	0
conv1d_8 (Conv1D)	(None, 1076, 64)	12352
batch_normalization_4 (BatchNormalization)	(None, 1076, 64)	256
max_pooling1d_4 (MaxPooling1D)	(None, 538, 64)	0
conv1d_9 (Conv1D)	(None, 536, 64)	12352
batch_normalization_5 (BatchNormalization)	(None, 536, 64)	256
max_pooling1d_5 (MaxPooling1D)	(None, 268, 64)	0
flatten_2 (Flatten)	(None, 17152)	0
dense_4 (Dense)	(None, 64)	1097792
dense_5 (Dense)	(None, 32)	2080
main_output (Dense)	(None, 1)	33

Total params: 1,125,825
Trainable params: 1,125,441
Non-trainable params: 384

Figure 7 : CNN model architecture

4. RESULT AND DISCUSSION

4.1 OUTPUT

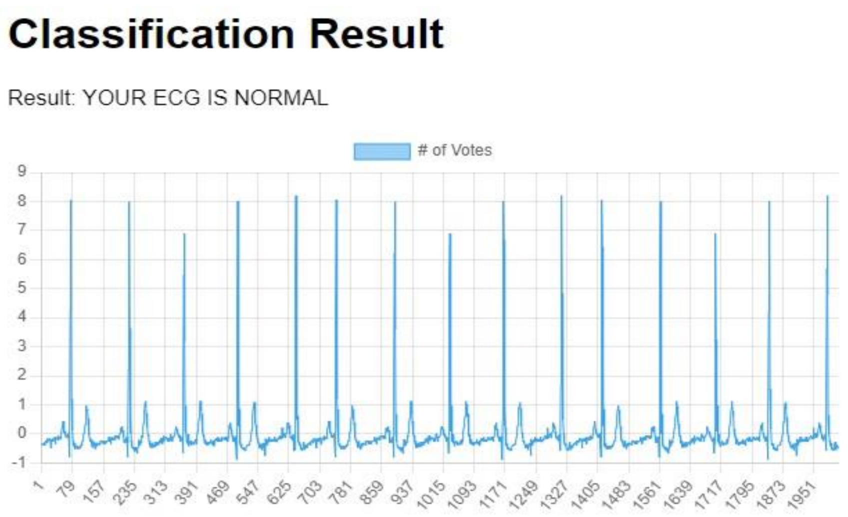

Figure 8 : Normal heart condition

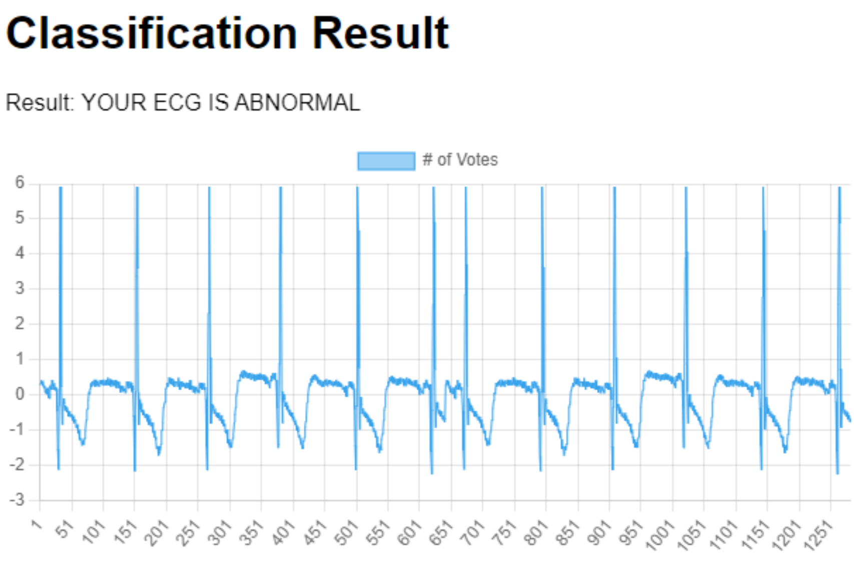

Figure 9 : Abnormal heart condition

We successfully trained a 1D CNN model for ECG classification and classified real time ECG data obtained by interfacing ESP32 and AD8232 sensor over the internet. For the classification of real time signal, we have deployed our model using Flask and created a frontend to display classification result and ECG signal in real time.

4.2 CLASSIFICATION REPORT

When an observation is both positive and predicted to be positive, it is referred to as a true positive (TP). When an observation is positive but is predicted to be negative, this is referred to as a false negative (FN). True Negative (TN) occurs when an observation is negative and is predicted to be negative. False Positive (FP) is characterized by a negative observation but a positive prediction. The total number of correctly classified positive examples divided by the total number of positive examples is defined as recall.

Recall (Sensitivity) = TP/ (TP + FN)

Precision is calculated by dividing the total number of positively classified examples by the total number of positively predicted examples. A high precision value indicates that an example labelled as positive is, in fact, positive.

Precision (Predictivity) = TP / (TP + FP)

The F -measure is a metric that represents both precision and recall. The F -measure will always be closer to the smaller among precision and recall values.

F1score = 2×Recall×Precision / (Recall+Precision)

Metric	**Value**
Precision	0.747
Recall	0.71
F1 Score	0.734

4.3 TRAINING LOSS CURVE

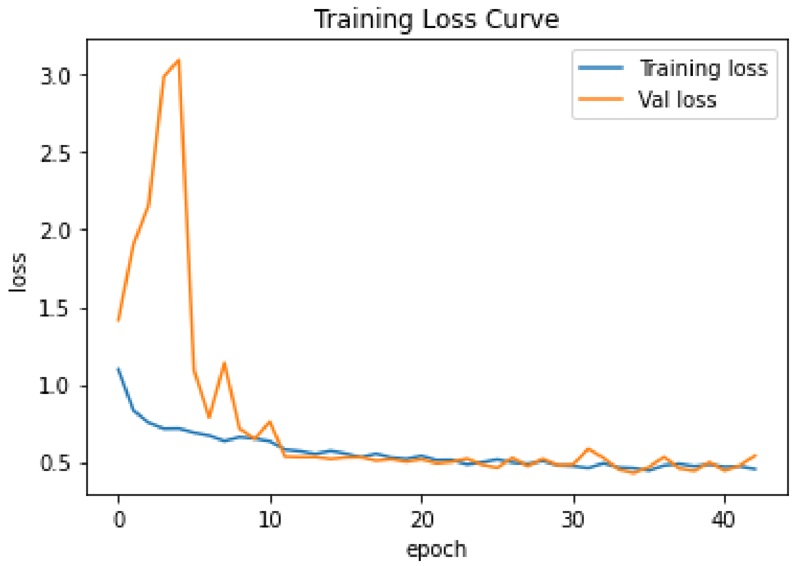

Figure 10: Training loss curve

4.4 CONFUSION MATRIX

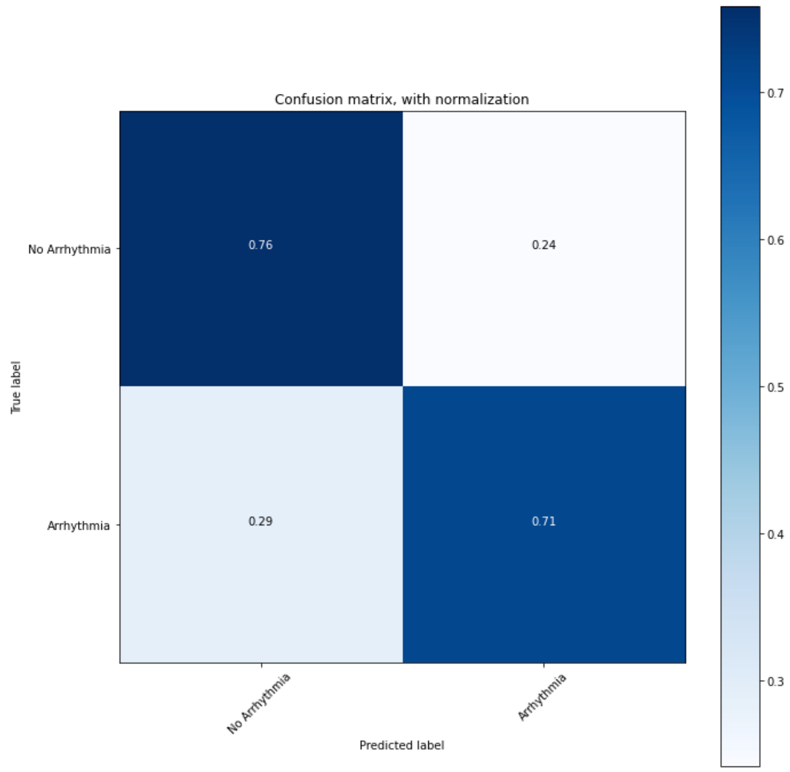

Figure 11 : Confusion matrix

4.5 ROC CURVE

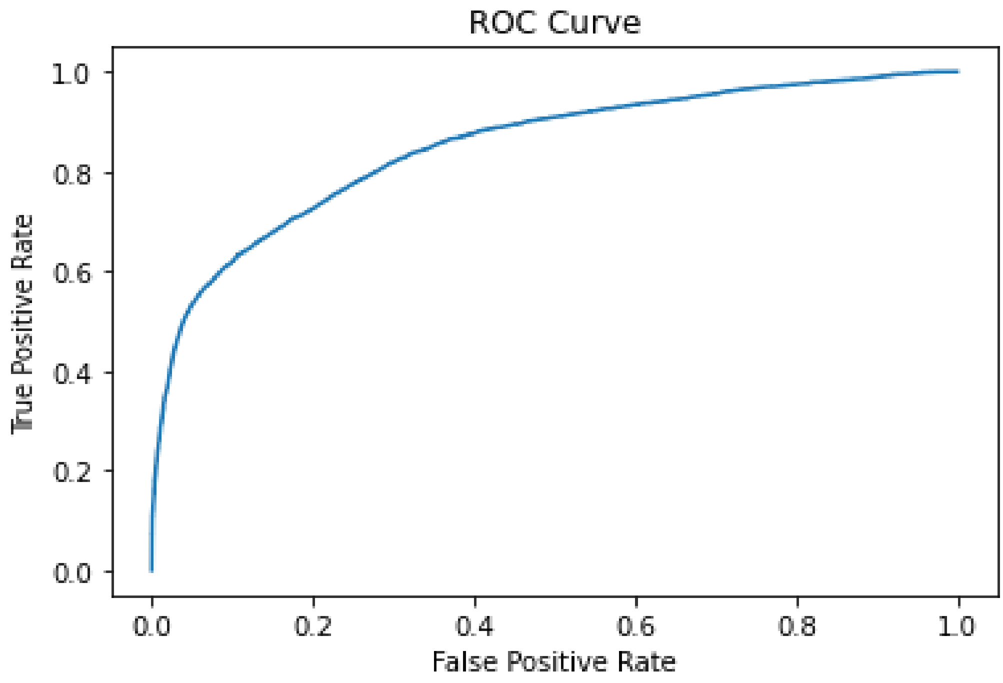

Figure 12: ROC curve

4.6 LIMITATIONS

- The performance of the model is highly dependent on the quality and size of the training dataset. If the dataset is small or contains noisy or inaccurate signals, the model may perform poorly.
- The ECG signal reading is highly affected by external noises such as electrical noise from power input of laptop, external magnetic field, etc.
- Reading is affected by the wrong placement of three leads of AD8232 sensor and movement of body.

4.7 PROBLEMS FACED

- MIT-BIH arrhythmia dataset has class imbalance which leads to biased prediction and lower accuracy in detecting abnormal ECG signals.
- While pre-processing ECG signal which involves filtering, re-sampling and segmenting the signals to extract relevant features, it was a very time consuming and heavy load task.
- Overfitting of the model was encountered in the training process.
- ECG signal from the sensor is affected by simple body movement, electrode motion artifacts and interference.

4.8 BUDGET ANALYSIS

S.N.	COMPONENTS	QUANTITY	COST(Rs)
1	ESP32	1	1000
2	AD8232 ECG SENSOR	1	2500
3	JUMPER WIRES	1	100
4	DISPOSABLE ECG ELECTRODE	5 pack	1500
5	CODING, DEBUGGING AND DOCUMENTATION	-	6000
6	TOTAL	-	11,100

5 CONCLUSION AND FUTURE ENHANCEMENT

5.1 CONCLUSION

- Our trained CNN model deployed to our website is successfully classifying the ECG signal in real-time acquired over the internet through interface of ESP32 and AD8232 sensor.
- Our model is comparable with the accuracy and efficiency of ECG arrhythmia classification of existing methods.
- Telemedicine is one possible use for the system, since it can allow fast identification and monitoring of abnormal cardiac events, supporting healthcare providers in providing prompt and preventative treatments.

5.2 FUTURE ENHANCEMENT

- Increase the number of classes: The current project is focused on classifying ECG signals into two classes - Normal and Abnormal. However, there are several types of abnormal ECG patterns that can be classified. By increasing the number of classes, the accuracy and reliability of the system can be improved.
- We can use industrial level ECG sensors like ADS1292R, MAX3003 etc. instead of Ad8232 sensors to have more accurate and precise results.
- By using high processing power microcontroller, we could be able to integrate the CNN model in the microcontroller and able to make portable device.
- WE can use 2-D image to classify the normality and abnormality of the ECG signal.
- We can use industrial level ECG sensors like CardioSecur, GE Healthcare MAC 5500 HD etc. for 12-lead ECG signal which can cover entire chest.

References

[1] A. I. Taloba, R. Alanazi, O. Shahin, A. Elhadad, A. Abozeid, A. El-Aziz and M. Rasha, "Machine algorithm for heartbeat monitoring and arrhythmia detection based on ECG systems," *Computational Intelligence and Neuroscience 2021,* 2021.

[2] how to electronics, "IoT ECG Monitoring with AD8232 ECG Sensor & ESP8266," how to electronics, 2022. [Online]. Available: https://how2electronics.com/iot-ecg-monitoring-ad8232-ecg-sensor-esp8266/. [Accessed 02 08 2022].

[3] M. Wu, Y. Lu and W. W. S. Y. Yang, "A Study on Arrhythmia via ECG Signal Classification Using the Convolutional Neural Network," *Frontiers in Computational Neuroscience,* vol. 14, 2021.

[4] F. J, R. V, C. G and P. E, "1-D Convolutional Neural Network for ECG Arrhythmia Classification," in *Progresses in Artificial Intelligence and Neural Systems*, A. Cecchi et al., 2021, pp. 269-279.

[5] X. Li, B. Cardiff and D. John, "A 1D Convolutional Neural Network for Heartbeat Classification from Single Lead ECG," in *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, 2020.

[6] L. Yeh, W. Chen, H. Chan, N. Lu, C. Wang, W. Twan, W. Du, Y. Huang, S. Hsu and T. Chen, "Integrating ECG Monitoring and Classification via IoT and Deep Neural Networks," *Biosensors (Basel),* vol. 11, no. 6, p. 188, 8 June 2021.

[7] A. Y. e. a. Hannun, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature medicine,* vol. 25, no. 1, pp. 65-69, 2019 Mar.

[8] Pallets, "Flask Documentation," 2022. [Online]. Available: https://flask.palletsprojects.com/en/2.2.x/?fbclid=IwAR3NBHqlOq24___ejOgHIfpOsmeIuDcFc8CqPcxVT0VFqJZmbZZhwm31YVw. [Accessed 11 March 2023].

[9] Analog Devices, "Analog," 2012-2020. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ad8232.pdf. [Accessed 1 8 2022].

[10] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. C. Ivanov and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals.," *Circulation [Online],* vol. 101, no. 23, pp. e215-e220, 2000.