ECG CLASSIFICATION USING DEEP LEARNING

PROJECT

ECE DUAL

Submitted by

Mohit Sipani **15MI414**

Ashish Kaushal **15MI415**

To

Dr. Philemon Daniel

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
HAMIRPUR-177005, HP (INDIA)

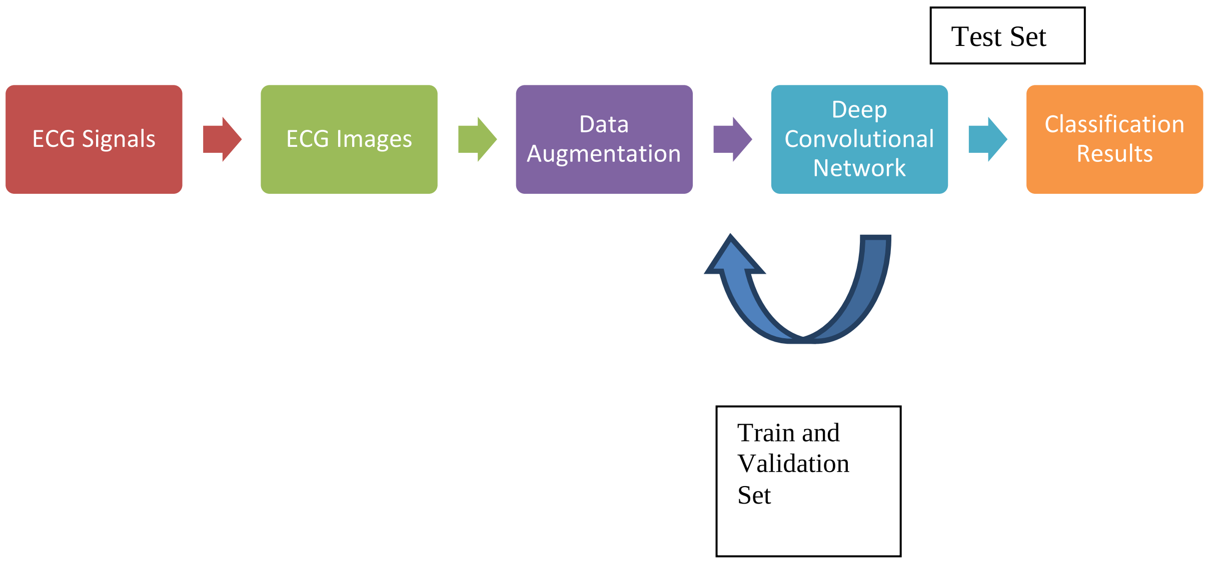
NOVEMBER 2019

Objective

To classify Electrography(ECG) signal using deep two-dimensional Convolutional Neural Network (CNN) with grayscale ECG images.

Methodology

ECG has been classified into seven categories, one being normal and the other six being different types of arrhythmia using deep two-dimensional CNN with grayscale ECG images. By transforming one-dimensional ECG signals into two-dimensional ECG images, noise filtering and feature extraction are no longer required. This is important since some of ECG beats are ignored in noise filtering and feature extraction. In addition, training data can be enlarged by augmenting the ECG images which results in higher classification accuracy.



Data Generation and Collection

I have used the MIT-BIH arrhythmia database for the CNN model training and testing. The MIT-BIH arrhythmia Database contains 48 half-hour ECG recordings, obtained from 47 subjects studied by the BIH arrhythmia laboratory between 1975 and 1979. For each record there are three files:

1. Annotation file

2. Signals file

3. Header file.

In this implementation, I have used the lead2 signals. The **get records ()** function in the code snippet below creates a list of all the records in the dataset. The **beat annotations ()** function finds the indices of the beats belonging to a particular category (In the code below, I have found the indices of Normal beats). The **segmentation ()** function is used to segment the beats of each category.

Processing & Network Layers

Since the CNN model handles two-dimensional image as an input data, ECG signals are transformed into ECG images during the ECG data pre-processing step. With these obtained ECG images, classification of seven ECG types is performed in CNN classifier step. The seven classes are: Atrial Premature Contraction, Normal, Left Bundle Branch Block, Paced Beat, Premature Ventricular Contraction, Right Bundle Branch Block and Ventricular Escape Beat.I have transformed ECG signals into ECG images by plotting each ECG beat. I first detected the R-peaks in ECG signals using **Biopsy** module of Python. Once the R-peaks have been found, to segment a beat, I took the present

R-peak and the last R-peak, took half of the distance between the two and included those signals in the present beat. Similarly, I did this for the next beat.

To convert these segmented signals into images, I have used Matplotlib and OpenCV. Since, in ECG signals, colours are of not any importance, I have converted them to grayscale images.

Output Layers

An 11-layer model has been used. The main structure of the model is very similar to VGGNet. The CNN model uses Xavier initialization for all the layers. For the activation function, we have used exponential linear units(ELU) unlike ReLU, that has been used in VGGNet.

For validation, I have used skLearn's train_test_split to divide the training and test set. There were 352295 images in the Training set and 44041 images in the Test set.

	Type	Kernel size	Stride	# Kernel	Input size
Layer1	Conv2D	3 x 3	1	64	128 x 128 x 1
Layer2	Conv2D	3 x 3	1	64	128 x 128 x 64
Layer3	Pool	2 x 2	2		128 x 128 x 64
Layer4	Conv2D	3 x 3	1	128	64 x 64 x 64
Layer5	Conv2D	3 x 3	1	128	64 x 64 x 128
Layer6	Pool	2 x 2	2		64 x 64 x 128
Layer7	Conv2D	3 x 3	1	256	32 x 32 x 128
Layer8	Conv2D	3 x 3	1	256	32 x 32 x 256
Layer9	Pool	2 x 2	2		32 x 32 x 256
Layer10	Full			2048	16 x 16 x 256
Layer11	Out			8	2048