Detection of Arrhythmia in Real-time using ECG Signal Analysis and Convolutional Neural Networks

Sashank Reddy, Surabhi B Seshadri, Sankesh Bothra G, Suhas T G, Saneesh Cleatus Thundiyil
Department of Electronics and Communication Engineering
BMS Institute of Technology and Management
(Affiliated to Visvesvaraya Technological University)
Bangalore, India
sashankr.official@gmail.com, surabhibs98@gmail.com, sankeshbothra@gmail.com, suhas.tg@outlook.com, saneesh@bmsit.in

***Abstract*—According to a study by the World Health Organization (WHO), cardiovascular diseases (CVD) are the leading cause of mortality across the world. Nearly 17.9 million deaths each year are due to CVDs, which equates to 31% of all deaths worldwide. An estimated 85% of these deaths are caused by heart attack or stroke. Cardiac Arrhythmia is a major disorder humans' experience which can lead to fatal conditions such as Cardiac Arrest or Stroke. Unless it is tested by a physician, one cannot discern if he/she has Arrhythmia because of its asymptomatic nature in some cases. An Electrocardiogram (ECG) is commonly used to check a patient's heart rhythm. In this paper, our focus is to present a method through which Cardiac Arrhythmia can be detected and classified in real-time by a wearable ECG device. The MIT-BIH Database is used to train and evaluate our CNN model, resulting in an accuracy of 99.625%.**

Keywords— Cardiac Arrhythmia, ECG Signal Analysis, Convolutional Neural Networks, Borderline-SMOTE, Wearable device.

I. INTRODUCTION

Cardiac Arrhythmia is a very common problem that tends to be asymptomatic. It occurs due to the malfunctioning of the electric pulses in the heart, resulting in an irregular heartbeat. The irregular heartbeat can be fast or slow as opposed to that of the normal heartbeat. The most severe case of Cardiac Arrhythmia presents itself when the irregular rhythms originate in the ventricles of the heart. This is known as Ventricular Arrhythmia (VA). The four well-known types of Cardiac Arrhythmia are Bradyarrhythmia, Supraventricular Arrhythmia, Ventricular Arrhythmia, and Premature or extra heartbeat. The changes in the pace of the heart are controlled by the sympathetic and parasympathetic nervous system. Overstimulation of the parasympathetic nervous system can lead to Bradyarrhythmia and overstimulation of the sympathetic nervous system due to increased stress can cause Tachyarrhythmia.

The test used to measure the electrical activity of the heart is the widely known ECG or the Electrocardiogram. It produces a graph of voltage versus time of the electric activity of the heart. An acute understanding of ECG Signals is usually used to determine the condition of Arrhythmia by medical professionals. In this paper, we have come up with an efficient and economical way to detect and classify Arrhythmia in an individual in real-time with the help of a wearable device using Convolutional Neural Networks. The wearable device contains a portable ECG sensor to record the user's heartbeats in real-time and a Wi-Fi module to save the heartbeats to the cloud. From the cloud, we run our CNN model on the saved data to detect and classify Arrhythmia.

II. RELATED WORK

S. Mousavi et al. present a novel approach to the classification of Arrhythmia, by using a Seq2seq network with bidirectional RNN units [1]. B. Pourbabaee et al. compare the various CNN structures as well as classifiers in a detailed manner [2]. Savalia, Shalin et al. demonstrate the correct way to build a model using Multilayer Perceptrons to attain the best results [3]. Pławiak Paweł et al. introduce a novel 3-layer architecture to tackle ensemble learning and deep learning simultaneously [4]. X. Fan et al. demonstrate the effectiveness of feature extraction from an ECG using MS-CNN [5]. S. M. Abubakar et al. design a small and efficient wearable ECG device using an FPGA [6].

Y. Xia et al. present a novel SDAE architecture along with a wearable device with Bluetooth connectivity for ECG classification [7]. Hui Han et al. display a new Borderline-SMOTE algorithm based on SMOTE for more effective oversampling of data [8]. M. Kachueeet al. illustrate use the of transfer learning to further improve a classification model, using the MIT-BIH and PTB Database [9]. Awni Y. Hannun et al. in their detailed study compare the diagnostic accuracies of neural network models with average cardiologists [10]. U. R. Acharya et al. devised a 9-stage CNN model for quick and accurate identification of arrhythmic heartbeats of patients [11].

III. NOVELTY

The paper is a coherent combination of hardware and software. Here, we have implemented real-time detection of Arrhythmia using convolutional neural networks. The hardware device captures the ECG signals from the patient which is in turn fed to a CNN model for the classification of the various categories of Arrhythmia. The main take away of this paper is that it not only has software to classify the category of Arrhythmia but also has a wearable device to perform this action in real-time which has not been implemented in the past with our level of accuracy. This accuracy is obtained by using state-of-the-art classification techniques coupled with over-sampling techniques for the over-sampling of minority classes in the highly imbalanced MIT-BIH Database.

978-1-7281-9617-6/20/$31.00 ©2020 IEEE.

IV. METHODOLOGY

A. Hardware implementation

We have used an Arduino Nano micro-controller that is based on ATmega328. The reason for selecting this micro-controller is due to its small form factor, coupled with good community support.

The AD8232 is a single-lead three-electrode ECG Heart Rate monitor used to obtain the electrical activity of the patient's heart. The electrodes are either connected close to the patient's heart or to their right arm, left arm and left leg to form an Einthoven's triangle. The electrical signals captured by the electrodes are initially received by the AD8232 and retransmitted to the Arduino Nano.

ESP8266 is a Wi-Fi module used to provide wireless data transmission access to the Arduino Nano and is used to connect it to the cloud. The hardware used is shown in Fig. 1. The visual output can be seen on the serial monitor on the Arduino IDE by setting the baud rate to 9600. The amplitude of the signals ranges from +300 to -200 about the average value of 500. The serial plotter on Arduino IDE shows the waveform corresponding to the values received.

B. Data reception and pre-processing

The electrodes are connected to the AD8232 breakout board through a 3.5mm jack. The AD8232 breakout board is connected to the Arduino Nano via jumper cables and further sent to the cloud with the help of ESP8266 for storage and processing. The sensor is configured to read approximately 200 samples per second. The data is then pre-processed to remove noise and distortions that may have been picked up during the recording procedure. The following steps are performed for pre-processing:

1) Filtering of data using Savitzky-Golay filters.
2) Normalization of data (ranging from 0 to 1).
3) Splitting of data into individual heartbeats.

Savitzky-Golay filters are great at removing added noise with a negligible change to the original properties of the signals being transformed. A Savitzky-Golay filter of polyorder = 2 and window size = 11 is used to eliminate noise while ensuring all key features of the ECG are not lost or changed.

Normalization of data is done to modify the amplitude of each ECG signal to vary between 0 and 1. This helps to provide our model with consistent data and thus helps improve prediction accuracies. Splitting of data is done using an in-house algorithm that takes into account the R-peak of a heartbeat in addition to the BPM of the user to accurately split each heartbeat without losing any key data points. The BPM can be calculated by the formula:

$$BPM = \textit{Sampling rate} \times 60 \,/\, (\textit{average number of samples between R-peaks}) \tag{1}$$

Each split heartbeat is then padded with zeroes until we obtain a sample with 500 data points. This padding is done as the CNN model below requires constant-sized input to be fed into it. Instead of resampling the data and having a small probability of losing some important data points, we have made the design choice to instead pad the signal so as to not risk the loss of such data points.

C. Convolution Neural Network Architecture

The architecture of the convolution neural network consists of 3 stages. The first stage takes in an input of size (500 x 1) and comprises of two 1-D convolution filters with a kernel size of 3 x 1 to which a Rectified Linear Unit (ReLU) non-linearity is connected. Then a one-dimensional Maximum Pooling layer is attached with a pool rate of 0.2. The last unit of this stage is a dropout layer with a dropout ratio of $\alpha = 0.1$. The second stage also has the same set of layers with a minor difference being the kernel size of the 1-D convolution filters being changed to 5. The third and final stage is again a copy of the previous two stages, with the 2 small changes being the 1-D convolution filter kernel size increased to 7, and the dropout layer ratio increased to 0.2. The CNN can be visualized as shown in Fig. 2.

The output parameters obtained at the end of the above 3 stages are fed to a series of 3 fully-connected dense layers. A densely connected layer provides learning features from all the combinations of features of the previous layer. The starting two dense layers are created with 64 units each, whereas the third and final dense layer is used as a softmax layer, which gives us the resultant vector matrix for each of the classes that the signal could belong to. The combined model contains 100,197 trainable parameters.

V. EXPERIMENTATION

The model as shown in Fig. 2 is trained and evaluated on the MIT-BIH Arrhythmia Database publically available on the 'physionet.org' website. It contains approximately 109,000 individual heartbeats from a total of 48 patients. The Database is run through the pre-processing algorithms to reduce the influence of noise on the model and to split each patient's ECG signals into individual heartbeats. Each heartbeat can belong to one of the five classes as shown in Table I.

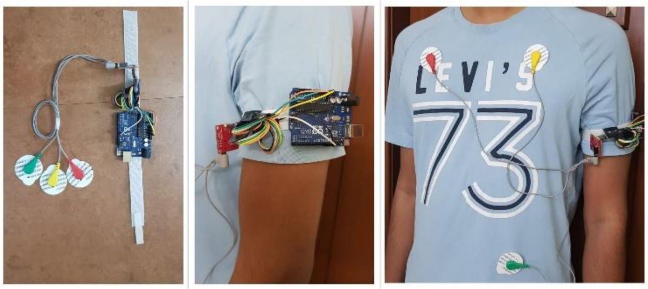

Fig. 1. Hardware assembly and connections. The leftmost image represents the wearable ECG device and the rightmost image shows us the connections on a user.

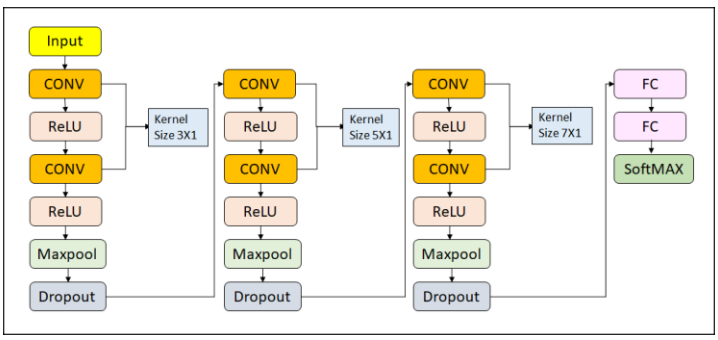

Fig. 2. CNN model Architecture and Flowchart.

Table I: CATEGORIES OF HEARTBEAT

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

Due to the Database being highly unbalanced, we obtained a total of 2,780 beats of category S and 803 beats of category F. When comparing this to the 90,623 beats of category N, we can assume the model is over-fitting. To counteract this, we use the Borderline-SMOTE algorithm to increase the number of beats in class S to 5,000 and beats in class F to 4,000. Borderline-SMOTE is a new and improved algorithm based on the SMOTE algorithm used in [1]. Borderline-SMOTE provides us with more realistic oversampled points and as mentioned by the authors in [8], also helps improve F1-value. This significantly improves our F1-scores and provides greater accuracy when classifying rare beats. After applying the Borderline-SMOTE algorithm to the Database, we obtain the dataset as shown in Table II.

The synthesized dataset from Table II is then split into training and testing data in the ratio of 90:10. The training dataset is further split into training and validation sets in the same 90:10 ratio. The CNN model is trained for a maximum of 200 epochs with the learning rate of $\alpha = 0.001$. We have noticed that beyond 30 epochs, the increase in accuracy is negligible and the validation loss begins to rise. This is a strong indication of the model being over fitted for the available data. Hence the EarlyStopping feature in Keras allows us to stop the training if we see no improvement in coming iterations. Adam Optimizer is the preferred optimization algorithm used here as it is very easy to implement, computationally extremely efficient and requires almost no memory for storage. The accuracy and loss graphs from the training are shown in Fig. 3.

Table II: DATASET AFTER THE APPLICATION OF BORDERLINE-SMOTE ALGORITHM

Class Distribution of Dataset	
Category	***Number of Samples***
N	90,623
S	5,000
V	7,229
F	4,000
Q	8,010

VI. RESULT

Table III shows us the difference in performance between our model and state of the art models proposed by other authors. The results obtained are only on the testing data samples which have not been previously used for the training of the model. We obtained an accuracy of 99.625% along with sensitivity, specificity and precision of 97.736%, 99.713% and 97.835% respectively with the confusion matrix as shown in Fig. 4. Table IV shows us the class-wise characteristics of our model.

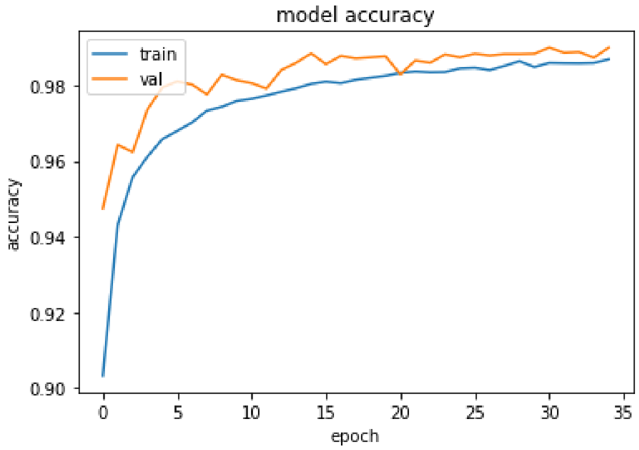

Fig. 3(a). Accuracy graph tracking both training and validation accuracies.

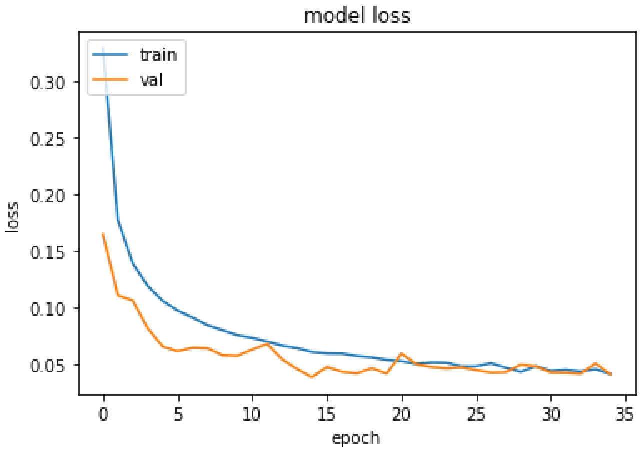

Fig. 3(b). Loss graph tracking both training and validation losses.

Table III: COMPARISON OF METRICS OF DIFFERENT MODELS

Algorithm	Method	Accuracy	Sensitivity	Specificity	Precision
Sashank et al.	Proposed Method	99.625	97.736	99.713	97.835
Pourbabaee B. et al. [2]	CNN with KNN	-	90.20	90.48	90.79
S. Shalin et al. [3]	MLP	88.7	-	-	-
X. Fan et al. [5]	MS-CNN	98.13	93.77	98.77	91.78
Kachuee M. et al [9]	Residual CNN	95.9	95.1	-	95.2
Acharya et al. [11]	9-layer CNN	94.03	96.71	91.54	97.86

Table IV: CLASS CHARACTERISTICS

Class	Accuracy	Sensitivity	Specificity	Precision	F1 Score	AUC
N	99.671	99.878	98.717	99.723	99.8	99.297
S	99.817	95.238	99.944	97.902	96.55	97.591
V	99.881	98.44	99.98	99.713	99.07	99.21
F	99.899	95.238	99.936	91.954	93.56	97.587
Q	99.982	99.884	99.99	99.884	99.88	99.937

Due to the light computational requirements of the model, we can run it on online servers with ease. The wearable device records and sends the ECG signals of the user to the server, and also classifies the individual beat and stores it in a cloud storage service for future reference. A final report of the user's session will be sent to them and if the user wishes to view a particular heartbeat, it will be viewed as shown in Fig. 5.

VII. CONCLUSION

We conclude that our proposed model outperforms most of the other proposed algorithms as seen in the literature. Our model performs better for use in the detection of minority classes due to the implementation of Borderline-SMOTE for oversampling. It is also significantly easier to run than most proposed algorithms due to its low computational requirements, as the model only has approximately 100,197 parameters with a maximum file size of 1.3 Megabytes. Lastly, we also provide a feature to the users to save and view their recordings from previous sessions, which could prove to be vital information in the future.

ACKNOWLEDGMENT

The authors of this paper acknowledge the department of Electronics and Communication Engineering at BMS Institute of Technology and Management, Bangalore for their co-operation and support in completing the project successfully. The authors would also like to express their gratitude to Visvesvaraya Technological University for funding the project and also considering it for the final round of evaluation of "AVISHKAR", a university-level project evaluation.

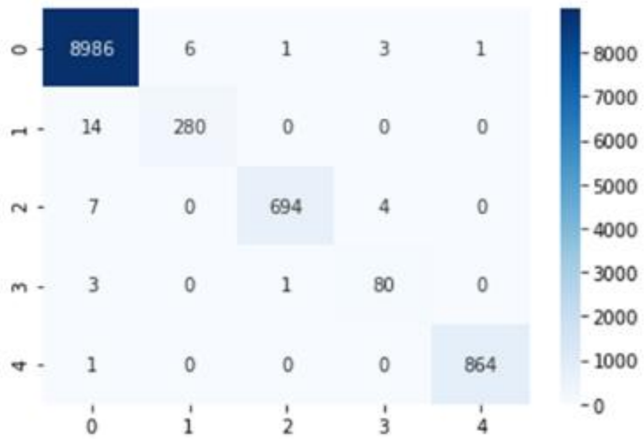

Fig. 4. Confusion Matrix of the model. Y-axis indicates the actual classes, whereas the X-axis indicates the class predictions made by the CNN model.

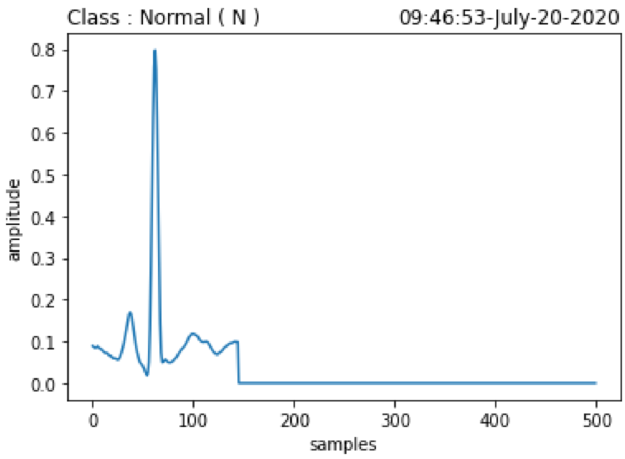

Fig. 5. An example of the predicted class of a single heartbeat. It indicates amplitude, duration of heartbeat and when it occurred with date-time.

REFERENCES

[1] S. Mousavi and F. Afghah, "Inter- and Intra- Patient ECG Heartbeat Classification for Arrhythmia Detection: A Sequence to Sequence Deep Learning Approach," ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, United Kingdom, 2019, pp. 1308-1312, doi: 10.1109/ICASSP.2019.8683140.

[2] B. Pourbabaee, M. J. Roshtkhari and K. Khorasani, "Deep Convolutional Neural Networks and Learning ECG Features for Screening Paroxysmal Atrial Fibrillation Patients," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 48, no. 12, pp. 2095-2104, Dec. 2018, doi: 10.1109/TSMC.2017.2705582.

[3] Savalia, Shalin & Emamian, Vahid. (2018). Cardiac Arrhythmia Classification by Multi-Layer Perceptron and Convolution Neural Networks. Bioengineering. 5. 35. 10.3390/bioengineering5020035.

[4] Pławiak, Paweł & Acharya, U Rajendra. (2019). Novel Deep Genetic Ensemble of Classifiers for Arrhythmia Detection Using ECG Signals. Neural Computing and Applications. 10.1007/s00521-018-03980-2.

[5] X. Fan, Q. Yao, Y. Cai, F. Miao, F. Sun and Y. Li, "Multiscaled Fusion of Deep Convolutional Neural Networks for Screening Atrial Fibrillation From Single Lead Short ECG Recordings," in IEEE Journal of Biomedical and Health Informatics, vol. 22, no. 6, pp. 1744-1753, Nov. 2018, doi: 10.1109/JBHI.2018.2858789.

[6] S. M. Abubakar, W. Saadeh and M. A. B. Altaf, "A wearable long-term single-lead ECG processor for early detection of cardiac Arrhythmia," 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, 2018, pp. 961-966, doi: 10.23919/DATE.2018.8342148.

[7] Y. Xia et al., "An Automatic Cardiac Arrhythmia Classification System With Wearable Electrocardiogram," in IEEE Access, vol. 6, pp. 16529-16538, 2018, doi: 10.1109/ACCESS.2018.2807700.

[8] Han, Hui & Wang, Wen-Yuan & Mao, Bing-Huan. (2005). Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. Adv Intell Comput. 3644. 878-887. 10.1007/11538059_91.

[9] M. Kachuee, S. Fazeli and M. Sarrafzadeh, "ECG Heartbeat Classification: A Deep Transferable Representation," 2018 IEEE International Conference on Healthcare Informatics (ICHI), New York, NY, 2018, pp. 443-444, doi: 10.1109/ICHI.2018.00092.

[10] Hannun, Awni & Rajpurkar, Pranav & Haghpanahi, Masoumeh & Tison, Geoffrey & Bourn, Codie & Turakhia, Mintu & Ng, Andrew. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature Medicine. 25. 10.1038/s41591-018-0268-3.

[11] Acharya, U Rajendra & Oh, Shu Lih & Hagiwara, Yuki & Tan, Jen Hong & Adam, Muhammad & Gertych, Arkadiusz & Tan, Ru San. (2017). A Deep Convolutional Neural Network Model to Classify Heartbeats. Computers in Biology and Medicine. 89. 10.1016/j.compbiomed.2017.08.022