Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 171 (2020) 524–531

Procedia Computer Science

www.elsevier.com/locate/procedia

Third International Conference on Computing and Network Communications (CoCoNet'19)

ECG Heartbeat Arrhythmia Classification Using Time-Series Augmented Signals and Deep Learning Approach

Pratik Kanani[a], Mamta Padole[b]

[a,b] *Department of Computer Science and Engineering, The Maharaja Sayajirao University of Baroda, Vadodara 390001, India*

Abstract

Electrocardiogram (ECG) signals are the best way to monitor the functionality and health of the cardiovascular system and also identify ailments related to it. Abnormal heartbeats are reflected in the ECG pattern and such abnormal signals are called as Arrhythmias. Automated classification and identification of the ECG arrhythmia signal that provides faster and more accurate result is increasingly becoming the need of the moment. Various machine learning skills have been applied to advance the accuracy of results and increase the speed and robustness of the models. A lot of focus has been given to the architectures and datasets employed but preprocessing of the data being equally important. In this paper, a preprocessing technique that significantly improves the accuracy of the deep learning models used for ECG classification is proposed with a modified deep learning architecture that adds to the training stability. With this preprocessing technique and deep learning model, the system is able to attain accuracy levels of more than 99% without overfitting the model.

Keywords: ECG; Arrhythmias; Augmentation; Deep Learning; Optimizer; transformations; stability; accuracy.

1. Introduction

Cardiac diseases are one of the major reasons of deaths around the world. Nearly one-third of all the deaths are caused by cardiac diseases [1]. Hence the accurate diagnosis of cardiovascular diseases is very essential [2-4]. Cardiologists and other doctors monitor the cardiac functioning and health using ECGs. An ECG is a record of the heart's electrical actions, which can be measured by placing electrodes on the skin. An ECG can be represented as a time-series data [16,37]. Until recent times, the analysis of ECG was done manually by medical practitioners. This comes with some drawbacks. Firstly, the analysis could be error prone. Secondly, it could be a very time-consuming process because of the presence of various types of waveforms and morphologies in the ECG signals that makes it difficult to classify those signals with precision. However, based on the available measures and PQRST points, one can classify them as Normal and Abnormal ECG signals [5].

Various techniques [6-8] of machine learning have been employed for analysis and classification of the ECG signals like multi-layer perceptron, support vector machines, random forests, decision trees etc. But all of these methodologies require a significant amount of pre-processing (e.g., band-pass filtering, convolution, etc.) to make the signal ready for use by the machine learning models. It is difficult to apply any of the classification and identification techniques on noisy signals.

* Corresponding author. Tel.: 9890917907.
E-mail address: pratikkanani123@gmail.com

1877-0509
10.1016/j.procs.2020.04.056

ECG bio signals are non-stationary signals and noisy [9]. Hence ECG signal need to be denoised-as noise suppresses the ECG signal and ECG interval characteristics [10]. Another major problem of the above-mentioned techniques is that the features need to be identified by a human or a cardiologist to correctly classify the ECG signals. This method evidently lacks the automated feature extraction principle which is more robust and reliable in terms of accuracy and scalability. An approach could be to a create feature representation of the dataset and the model is then trained.

Deep learning architectures provide the flexibility of learning the features on their own and moreover only learn the features that are relevant to the identification tasks of the signals at hand [11-14]. These architectures have a large number of variables that need to be trained and the training requires a lot of computation power. But the availability of high computation power, cloud computing has eliminated the training related issues.

2. Datasets

Here, the labeled MIT - BIH Arrhythmia Dataset for supervised learning is used [15-17]. This dataset contains ECG signals from five classes, namely: 'N'- 0, 'S'- 1, 'V'- 2, 'F'- 3, 'Q'- 4. The ECG signals from these classes have the properties as given below. The ECG signals in this dataset are represented at sampling frequency of 125 Hz with a total of 1,09,446 samples encompassing the five aforementioned classes. This dataset has widely been used in arrhythmia classification using Deep Learning constructions. Here, it is tried with different techniques and modifications in architectures to improve the accuracy. The samples of this dataset resemble to the ECG shapes of heartbeats that are normal as well as those affected by different types of myocardial infarctions and arrhythmia. The signal samples are preprocessed and segmented such that each segment corresponds to a heartbeat. A mapping between the beat descriptions and AAMI EC57 categories [18] is written below. The 80% of dataset is used for training purpose and the 20% is for the testing purpose.

'N' Category shows Normal, Left or Right bundle branch block, Atrial escape and the Nodal escape
'S' Category is for signals like Atrial Premature, Aberrant atrial premature, Nodal premature, and Supra-ventricular premature
'V' Category is showing Premature ventricular contraction and Ventricular escape
'F' Class is the label for Fusion of ventricular and normal whereas
'Q' label is for Paced, Fusion of paced and normal Unclassifiable

This dataset comprises of the ECG samples from 47 subjects originally noted at a sampling frequency of 360 Hz and later down sampled to 125 Hz for our purpose. All the samples of the dataset are then augmented. When machine learning model is getting trained, it is actually tuning its parameters so that it can map inputs (sample) to the outputs (class label). To have lots of parameters, machine learning model needs lots of data, to get properly trained and thus, if parameters are adjusted in the right way then it leads to reduced model loss. Multiple augmentation techniques are present like rotation of images, flipping of images, cropping, scaling and translating the images [19]. Time series data comprises of stretching, amplifying, stretching and amplifying and squeezing the samples along the required axes. In this paper, to perform the time series operations [9] on the signals, the numpy [20] library and different functions like squeeze(), stretch(), amplify, and augment() are used. The aforementioned pre-processing done on the original dataset is elaborated in section 3.1.

3. Methodology

3.1. Pre-processing

Normal ECG signals are a series of peaks consisting of a series of individual waves namely: T wave, QRS wave, P wave. A series of normal ECG signals looks as given in Figure 1.

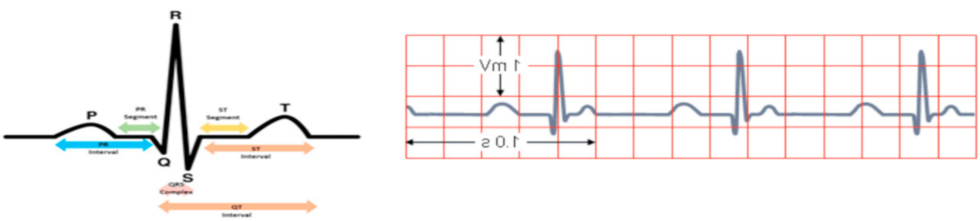


Fig. 1. Normal ECG wave with PQRST points and different intervals. [10] [21]

The dataset consists of individual signals from one R peak to the next R peak in the series for five different classes. In the preprocessing phase each signal is taken and the following set of transformations are performed:

- Squeeze - resamples the signal along the time (horizontal) axis by shortening the distance between two consecutive samples.

- Stretch - resamples the signal along the time (horizontal) axis by elongating the signals.
- Amplify - amplifies the signal (along vertical axis) but still keeps the range between 0 and 1.
- Shrink - reduces the amplitude (along vertical axis) as well as squeezes the signal.

These transformations are applied to every signal of the dataset and each of these transformations is saved and appended to the original dataset thereby summing to 5,47,230 samples in the entire dataset, contrary to the dataset used in [22]. These transformations in the signals are completely lossless transformations [23] and do not change the nature, quality and file size of the signals. In Deep Learning, each layer learns certain functions which can be extracted by our deep learning model [24].

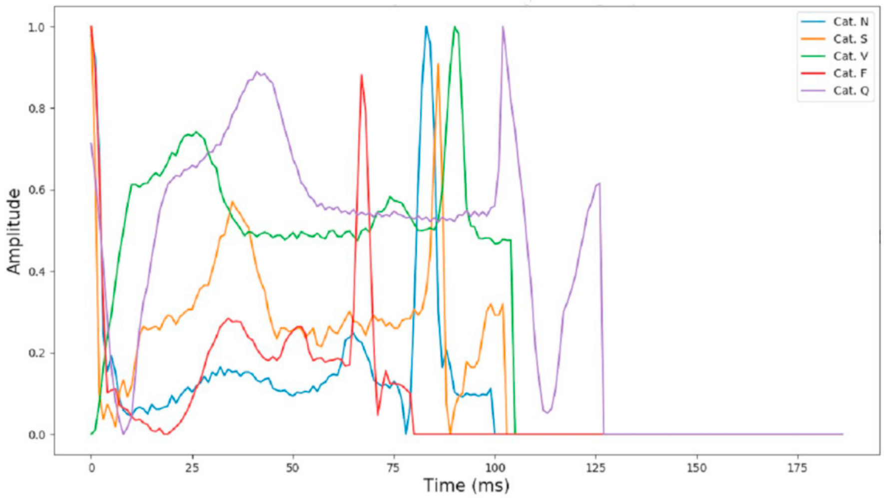


Fig. 2. ECG signals from all five categories.

In figure 2, the ECG signal waveform from all five categories are drawn and are shown by different colors, e.g., the green color wave shows the ECG signal related to ‘V’ class which is for Premature ventricular contraction and Ventricular escape.

In figure 3, One sample of ECG waveform has been taken and augmentation is performed on it as follows.

- Blue colored wave is normal wave
- Orange wave is stretched in X axis direction
- Green wave is squeezed in X axis direction
- Red wave is amplified in Y axis direction
- Purple wave is shrinked in X and Y axes direction

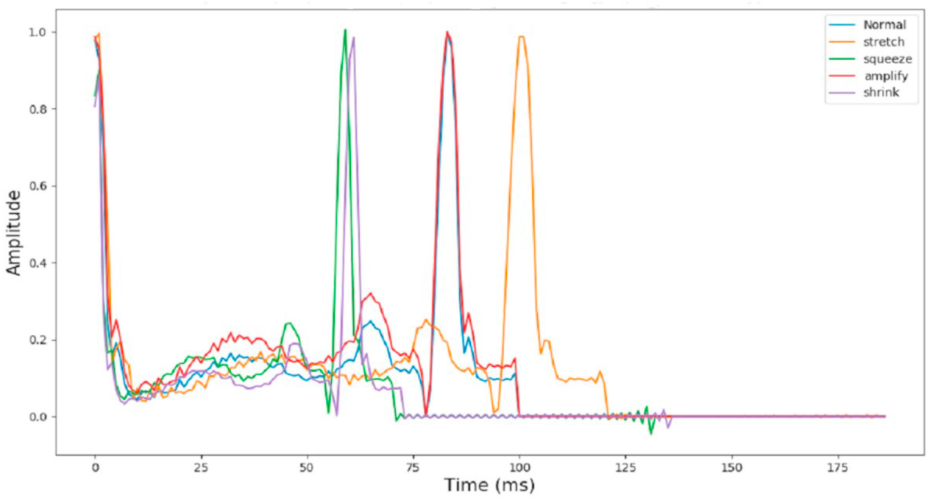


Fig. 3. Four transformations on a single sample.

3.2. Deep Learning Model

Artificial Intelligence aids in Intelligent behavior of computer systems. In Machine Learning the program is capable to learn itself but the learning features should be correctly identified and trained by the programmer. Whereas the Deep Learning uses more mathematics, neural networks and computation that enables computer itself to identify better features for generating accurate and useful results [34].

The model used in [22] applied 1D convolution along the time axis and each layer had 32 filters with kernel size 5. In this paper a model to train the augmented dataset is proposed. Moreover, in the proposed model a kernel size of 64 is used. It has empirically been proven [25] to provide convergence stability to the model and also extracts more number of features because of a higher number of filters. The kernel size can decide the number of input features required to be combined with number of new output features. A max pooling layer of kernel size 5 and stride length of two is also deployed. The model comprises of six residual blocks that are taken by two fully connected layers, each of them having 32 neurons and a softmax activation layer that predicts the output class probabilities. Each of the residual block consists of two convolutional layers, two ReLU [26-27] activation layers and a max-pooling layer. ReLU is Rectified Linear Unit which is activation or transfer function. It helps neural network to decide the output in terms of yes and no by mapping output to some values like 1 and 0 or -1 and 1, depending on the model function. Max pooling layer down samples the input and reduces number of dimensions and allows to select maximum value [28]. The model consists of 15 weight layers as contrary to the model in [22] that has 13 weight layers.

The output of the first convolutional layer is added to the output of the third convolutional layer. These residual blocks or ResNet [35] blocks contain most of the extracted features in the model. Having 6 residual blocks increases the number of features extracted as well as increasing the accuracy without overfitting the model. The output is received after the softmax layer. Figure 4 depicts proposed model.

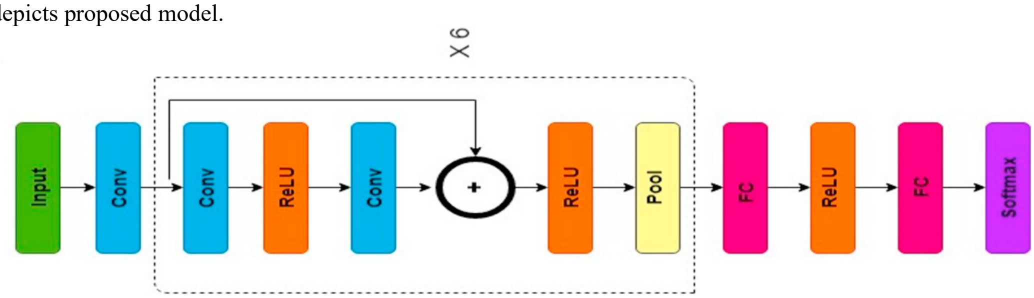


Fig. 4. Proposed Architecture.

3.3. Implementation

Keras [29] and Scikit-learn [30] for Python 3.6 [31] libraries for the deep learning related computation are used. The model has been trained on Google Colaboratory [32], wherein each model experiment is trained for 120 epochs. The loss function is the softmax output from cross entropy loss. The model uses Adam optimizer [33] with learning rate, beta-I, beta-II as 0.001, 0.9, 0.999 respectively and the learning rate decays exponentially by a factor of 0.75 every 10,000 iterations.

4. Results

Experiments were performed on the proposed model as well as the model used, as discussed by Kachuee et. al., in paper [22] using our augmented dataset as well as the original dataset [36] used in [22]. Four sets of results have been obtained, namely: initial model [22] with original dataset, initial model with augmented dataset, new proposed model with original dataset and new proposed model with augmented dataset.

Table 1. A table showing the results of experiments.

Model	Dataset	Ranking based Average precision	f1-score	Weighted Ranking Loss	Coverage error
Initial model	original dataset	0.9397	0.89	0.0399	1.1275
Initial model	augmented dataset	0.9906	0.98	0.0048	1.0192
Proposed model	original dataset	0.9474	0.90	0.0319	1.1595
Proposed model	augmented dataset	0.9912	0.98	0.0047	1.0190

The augmented dataset evidently provides a greater accuracy over the validation dataset and also reduces the ranking loss by a factor of 1/10. Now, the model only provides a marginal increase in accuracy as compared to the model used in [22] but the plots of the model's losses and accuracies in Figure 5 and 6 show that the proposed model achieves convergence faster and has more training stability than the previous model and thus making the model more accurate. It is also evident from the plots that the augmented dataset provides better validation accuracy. F1 score is the balance between precision and recall. Coverage [38] is the proportion with which the classifier makes predictions with the given dataset. Ranking defines the overall relative score of all items in learning rather than individual item scores [39].
It is observed that the f1 value of the initial model with initial data set is 0.89, while the same model with augmented dataset gives the value around 0.98 which shows augmentation helps in improving the prediction accuracy. The proposed model gives 0.98 without augmentation while with augmentation it also gives f1 score as 0.98, but in this last case the ranking loss and coverage error value is less. This makes the proposed model more accurate.

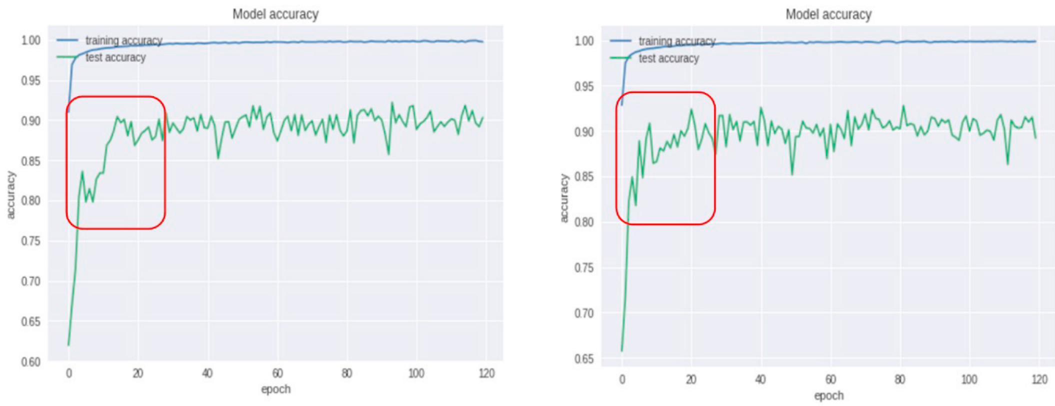


Fig. 5. Model Accuracies for the defined model and the proposed model trained on the original Dataset

In figure 5 the testing accuracy of proposed model is more stable than the testing accuracy of defined model for first 20 epochs. It is shown by the red rectangle.

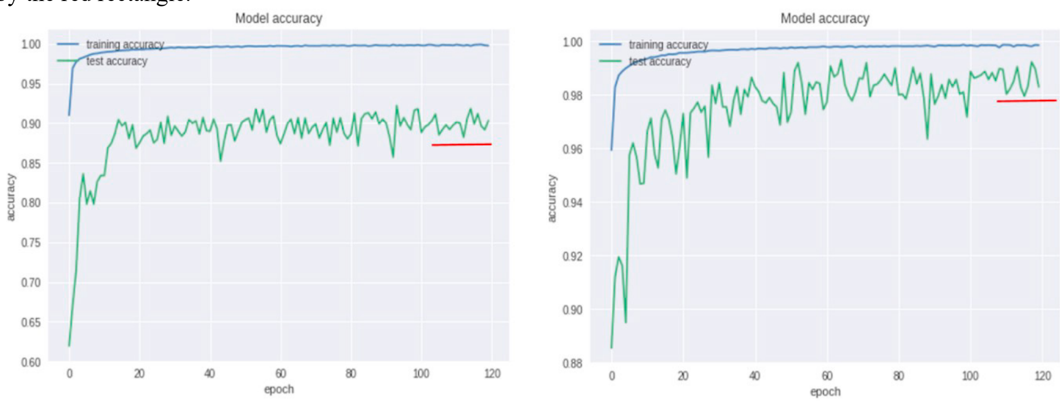


Fig. 6. Model Accuracies for the defined model and the proposed model trained on the augmented Dataset

In figure 6 the testing accuracy of proposed model is not that stable as of defined model but the final accuracy values are higher and is more than 98, whereas the defined model has around the range of 90, as shown by red lines.

In figure 7, The categories 'F' and 'S' are not predicted that correctly as of other classes as they have very minute distinguishes in the shapes of respective waveforms of 'N' class.

In figure 8, The categories 'F' and 'S' are now predicted very well though they look similar to 'N' class.

The four confusion matrices show the test predictions of the samples into the five categories defined in the dataset, namely; 'N', 'S', 'V', 'F' and 'Q'. The accuracies depicted in the confusion matrices in figure 8 are much better than the ones depicted in the figure 7 which prove that the augmented dataset does help achieve better results.
The data set used for the proposed model is more than 5.5 lakhs after the augmentation of original dataset. Also, from the accuracy graphs it is evident that the new model adds stability to the model.

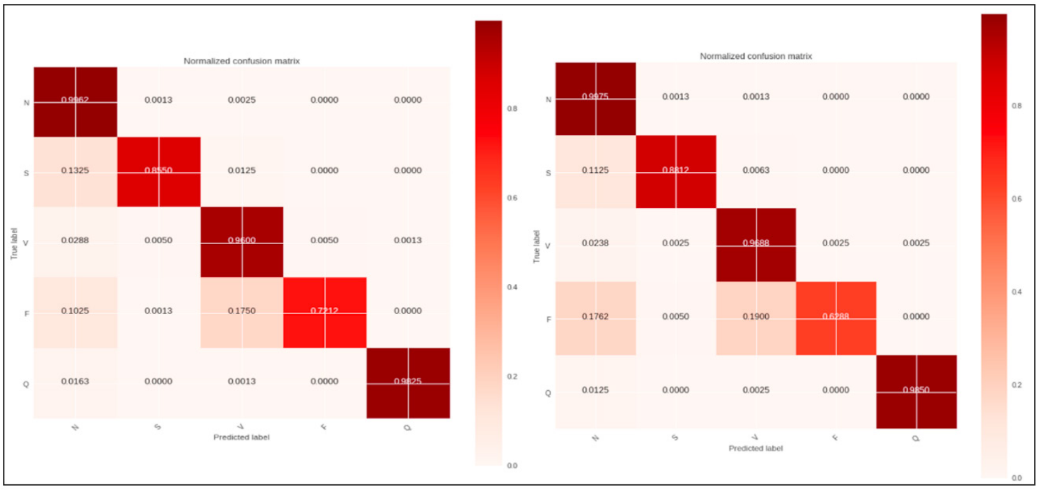


Fig. 7. Confusion matrix for the defined model and for the proposed model using original dataset.

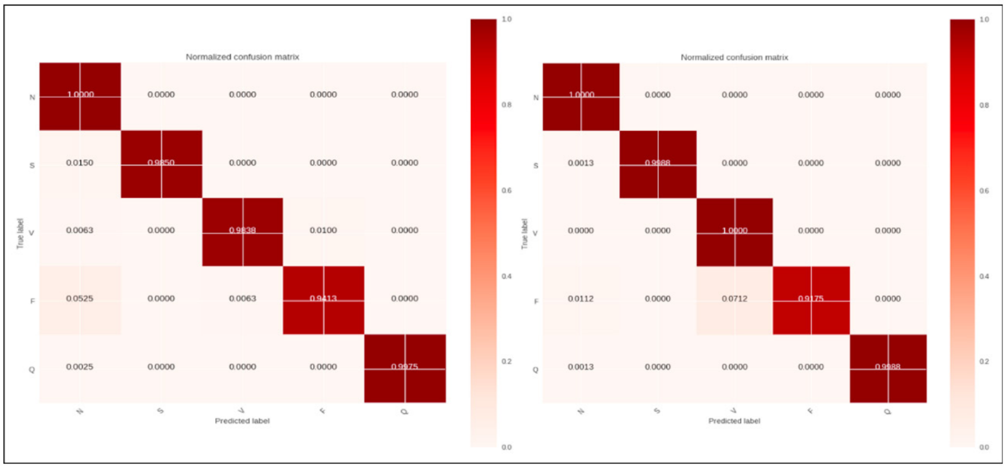
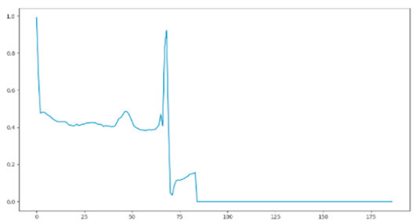


Fig. 8. Confusion matrix for the defined model and for the proposed model using augmented dataset.

5. Test Cases

To test the system, two ECG signals are recorded at real time. One is normal and the second is abnormal [5]. The inputs are given to the system to observe its behavior. The system outputs five numbers which are the five probabilities values with which signal belongs to a particular class.



```
[[0.29393518 0.05480201 0.5598804  0.07608867 0.01529378]]
Abnormal!
```

Fig. 9. An abnormal ECG beat is classified to third class i.e., Label ‘V’ with 0.5598 chances of probability.

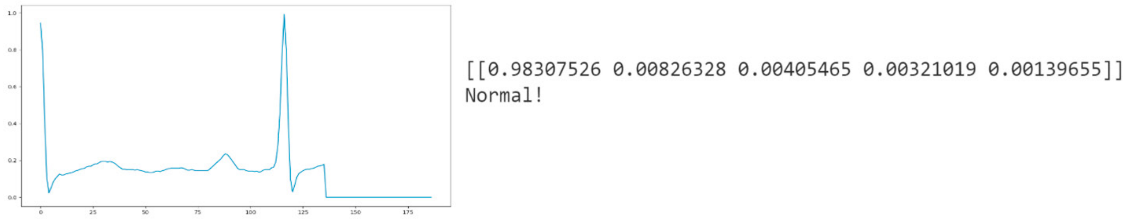


Fig. 10. A Normal ECG beat is classified to First class i.e., Label 'N' with 0.9830 chances of probability.

6. Conclusion

Applying augmentations to the dataset can not only make the model training more accurate but also stabilize it at higher accuracies. The proposed model consists of 6 residual blocks which means there is scope of overfitting the data but the augmented dataset also prevents overfitting by making classification difficult in the testing phase. The proposed model still displays high accuracy in such conditions. Thereby depicting its caliber to make highly accurate predictions with an accuracy rate of 99.12%.

7. Future Scope

In the future, this model can be improvised to make predictions for a larger amplitude and frequency ranges depicting cardiac disorders and arrhythmia. The current system classifies the ECG arrhythmia signals by using a model that has been trained by signals having values in the range 0-1 and all signals are recorded at 125 Hz. Each ECG measuring device has their own specified sequences and output voltage range, for example the Arduino will give the same signal amplitude in the range of 0-1023. So, the current system may not work for signals having different amplitudes and frequency ranges.

References

[1] World Health Organization, http://www.who.int/mediacentre/factsheets/ fs317/en/.

[2] Heart diseases and disorders, https://www.hrsonline.org/Patient-Resources/Heart-Diseases-Disorders

[3] A. Esmaili, M. Kachuee et al., "Nonlinear cuffless blood pressure estimation of healthy subjects using pulse transit time and arrival time", IEEE Transactions on In-strumentation and Measurement, vol. 66, no. 12, pp. 3299–3308, 2017.

[4] A. Dastjerdi, M. Kachuee et al., "Non-invasive blood pressure estimation using phonocardiogram", in Circuits and Systems 2017 IEEE International Symposium on. IEEE, 2017, pp. 1– 4.

[5] Kanani P., Padole M. (2018) Recognizing Real Time ECG Anomalies Using Arduino, AD8232 and Java. In: Singh M., Gupta P., Tyagi V., Flusser J., Ören T. (eds) Advances in Computing and Data Sciences. ICACDS 2018. Commu-nications in Computer and Information Science, vol 905. Springer, Singapore

[6] O. Inan, L. Giovangrandi et al., "Robust neural network-based classification of prem-ature ventricular contractions using wavelet transform and timing interval features", IEEE Transactions on Biomedical Engineering, vol. 53, no. 12, pp. 2507–2515, 2006.

[7] O. Sayadi, M. Shamsollahi, et al., "Robust detection of premature ventricular contrac-tions using a wave-based bayesian framework," IEEE Transactions on Biomedical Engineering, vol. 57, no. 2, pp. 353–362, 2010.

[8] M. Kachuee, M. M. Kiani et al., "Cuffless blood pressure estimation algorithms for continuous health-care monitoring", IEEE Transactions on Biomedical Engineering, vol. 64, no. 4, pp. 859–869, 2017.

[9] Time Series Analysis in Python, https://www.machinelearningplus.com/time-series/time-series-analysis-python/

[10] The student physiologist, https://thephysiologist.org/study-materials/the-normal-ecg/

[11] U. Acharya, S. Oh et al., "A deep convolutional neural network model to classify heartbeats", Computers in Biology and Medicine, vol. 89, pp. 389–396, 2017.

[12] S. Kiranyaz, T. Ince et al., "Real-Time Patient-Specific ECG Classification by 1-D Convolutional Neural Networks", IEEE Transactions on Biomedical Engineering, vol. 63, no. 3, pp. 664–675, 2016.

[13] L. Jin and J. Dong, "Classification of normal and abnormal ECG records using lead convolutional neural network and rule inference", Science China Information Scienc-es, vol. 60, no. 7, 2017.

[14] P. Rajpurkar, A. Hannun et al., "Cardiologist-level arrhythmia detection with convo-lutional neural networks", arXiv preprint arXiv:1707.01836, 2017.

[15] A. Goldberger, L. Amaral et al., "Physiobank, physiotoolkit, and physionet", Circula-tion, vol. 101, no. 23, pp. e215–e220, 2000.

[16] G. Moody and R. Mark, "The impact of the mit-bih arrhythmia database", IEEE En-gineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45–50, 2001.

[17] R. Bousseljot, D. Kreiseler et al., "Nutzung der ekgsignaldatenbank cardiodat der ptb uber das internet", ¨ Biomedizinische Technik/Biomedical Engineering, vol. 40, no. s1, pp. 317–318, 1995.

[18] A. for the Advancement of Medical Instrumentation et al., "Testing and reporting per-formance results of cardiac rhythm and ST segment measurement algorithms," ANSI/AAMI EC38, vol. 1998, 1998.

[19] Data Augmentation, https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced

[20] Python numpy, https://www.programcreek.com/python/example/50976/numpy.random.uniform

[21] The basis of ECG Diagnosis, http://www.bem.fi/book/19/19.htm

[22] M. Kachuee, S. Fazeli et al., “ECG Heartbeat Classification: A Deep Transferable Representation”, https://arxiv.org/pdf/1805.00794

[23] O. B’charri, R. Latif et.al., “An Efficent Lossless Compression Scheme for ECG Signal”, International Journal of Advanced Computer Science and Applications, Vol. 7, No. 7, pp.210-215, 2016

[24] What is Deep Learning? , https://in.mathworks.com/discovery/deep-learning.html

[25] Intuitively Understanding Convolutions for Deep Learning, https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1

[26] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann ma-chines”, in Proceedings of the 27th international conference on machine learning, 2010, pp. 807–814.

[27] Activation Functions in Neural Networks, https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

[28] Max Pooling Layer, https://computersciencewiki.org/index.php/Max-pooling_/_Pooling

[29] Keras, https://github.com/keras-team/keras/

[30] Scikit-learn, https://github.com/scikit-learn/scikit-learn

[31] Python 3.6, https://www.python.org/downloads/release/python-360/

[32] Google Colaboratory, https://colab.research.google.com/

[33] Gentle Introduction to the Adam Optimization Algorithm for Deep Learning, https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/

[34] AT vs. ML vs. DL, https://skymind.ai/wiki/ai-vs-machine-learning-vs-deep-learning

[35] Residual Blocks, https://towardsdatascience.com/residual-blocks-building-blocks-of-resnet-fd90ca15d6ec

[36] ECG Heartbeat Categorization Dataset, https://www.kaggle.com/shayanfazeli/heartbeat

[37] G. B. Moody, R. G. Mark and A. L. Goldberger, "PhysioNet: Physiologic signals, time series and related open source software for basic, clinical, and applied re-search," 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Boston, MA, 2011, pp. 8327-8330.

[38] Glossary of Terms, ai.stanford.edu/~ronnyk/glossary.html

[39] Learning to Rank, https://medium.com/@nikhilbd/intuitive-explanation-of-learning-to-rank-and-ranknet-lambdarank-and-lambdamart-fe1e17fac418