THE UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

THE GRAINGER COLLEGE OF ENGINEERING

Deep Learning for Computer Vision (CS 444)

Project Report
Under the guidance of Prof. Svetlana Lazebnik

Project By:

S. No.	UIN	Name	NetID
1.	658266911	Riya Saini	riya6
2.	678182751	Yasomitra Sampat Kota	ykota2

INTRODUCTION

ECG signals are extensively used for diagnostic purposes such as measuring heart rate, emotion recognition, and most importantly detecting heart diseases by identifying heart abnormalities [1]. While these kinds of analyses have traditionally been performed by healthcare professionals, in recent times, machine learning algorithms have become a popular method. With the surge in discovery and development of novel neural networks, the advent of deep learning techniques in the field of biomedical sciences has increased.

According to the literature, deep learning models are widely used for accurate and efficient diagnostic using ECG signals. Their ability to learn highly complex patterns allows them to produce results that surpass classifications provided by experts [2]. Generally, the algorithms have been time-consuming and complex due to the involvement of feature extraction and pre-processing which is required to extract useful information from noisy ECG signals. In this project, we have implemented and validated Attention-based Convolutional Neural Network (ABCNN) for ECG based heart diseases detection [3]. In this paper, the authors aim to leverage multi-head attention along with CNN to extract features from the raw ECG signal. ABCNN bypasses the denoising and other pre-processing characteristic of algorithms used for ECG signal analysis and outperforms existing baseline models [3].

In this project, we had three distinct goals— (i) implement the ABCNN architecture and reproduce the reported results using the dataset used in the source, (ii) validate the performance of the model using different datasets, and (iii) Perform transfer learning. The original dataset consisted of ECG records of 48 patients, with around 65000 data points captured every 800 milliseconds. The final data was obtained after sampling individual beats and using appropriate transformations, making it suitable for the ABCNN network. The model contains and *input layer* which receives the raw ECG signal, followed by *multi-head attention layers* which is responsible for assigning weights to the incoming slices of data. These layers are followed by *convolutional* and *maxpooling layers* for feature extraction. The final output is generated by *fully connected layer* which uses *softmax* as the activation function [3].

After training the model on the primary dataset, as described above, different datasets were used to validate the performance of the model. The aim was to check the ability of the model in learning highly complex and closely related ECG signal. Two datasets were used— (i) ECG signals with subclasses of Normal heartbeat as seen in the primary dataset (VD I) [3], and (ii) ECG signals capturing myocardial infraction (VD II) [4]. The ABCNN model was trained on the datasets and performance was evaluated. An AUC score of 0.9983 and 0.9902 was obtained for VD I and VD II. AUC score has been used as an evaluation metric since it is suitable for a highly imbalanced data. Other evaluation metrics such as precision and recall have not been reported since they are highly dependent on the number of samples [3]. In addition to training ABCNN from scratch, we also attempted to performed transfer learning. The model trained on the primary dataset was fine-tuned to classify the ECG signals of the other two datasets. However, it was observed that transfer learning was less accurate compared to training the complete model.

The following sections of the report highlight the methodology adopted as outlined above. The data pre-processing algorithm has been explained and the validation process is discussed in detail. The results section elucidates the performance of the model and the various insights we gathered. The report is finally concluded with a brief conclusion and future scope. The source code from which the project is referenced is available in Reference. 3.

Methodology

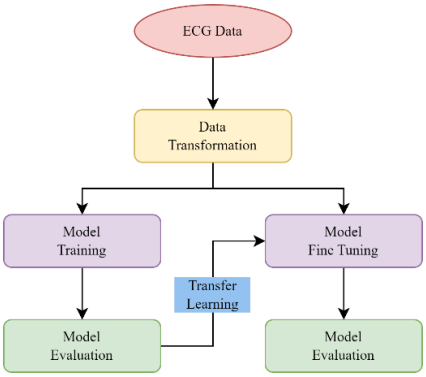


Figure 1: Project flow

1. ECG Data

The first step was to identify potential ECG datasets that could be used for the validation of ABCNN. The primary dataset from the original paper captured five different arrhythmia conditions. Two more ECG datasets apart from the primary were used for validation.

The first validation dataset was a subsection of the primary dataset. As per the original paper, each primary class has sub-categories. One primary class was chosen and was trained to see if the model can distinguish between the subcategories. The arrhythmia dataset had 5 primary categories and subcategories as shown in Table I.

The other dataset captured was myocardial infarction (MI) [4]. The MI dataset consisted of two classes, indicating patients with the presence or absence of the abnormality. All the samples in this dataset were cropped, down sampled, and padded with zeroes [4].

Table I: Classes in the primary dataset

Category	Sub-Category	Annotation
N	N, L, R, e, j	Normal beat
V	V, E	Premature ventricular contraction
S	A, a, J, S	Supraventricular premature beat
F	F	Fusion of ventricular and normal beat
Q	/, f, Q	Unclassifiable beat

2. Data Transformation

For the primary dataset, each patient's ECG records were sampled into lengths of 480 beats, and the category that fell in between the sample was assigned to that beat correspondingly. This resulted in around 100,000 samples in total. They were later transformed into a shape of $[2, 240]$ [3]. As shown in

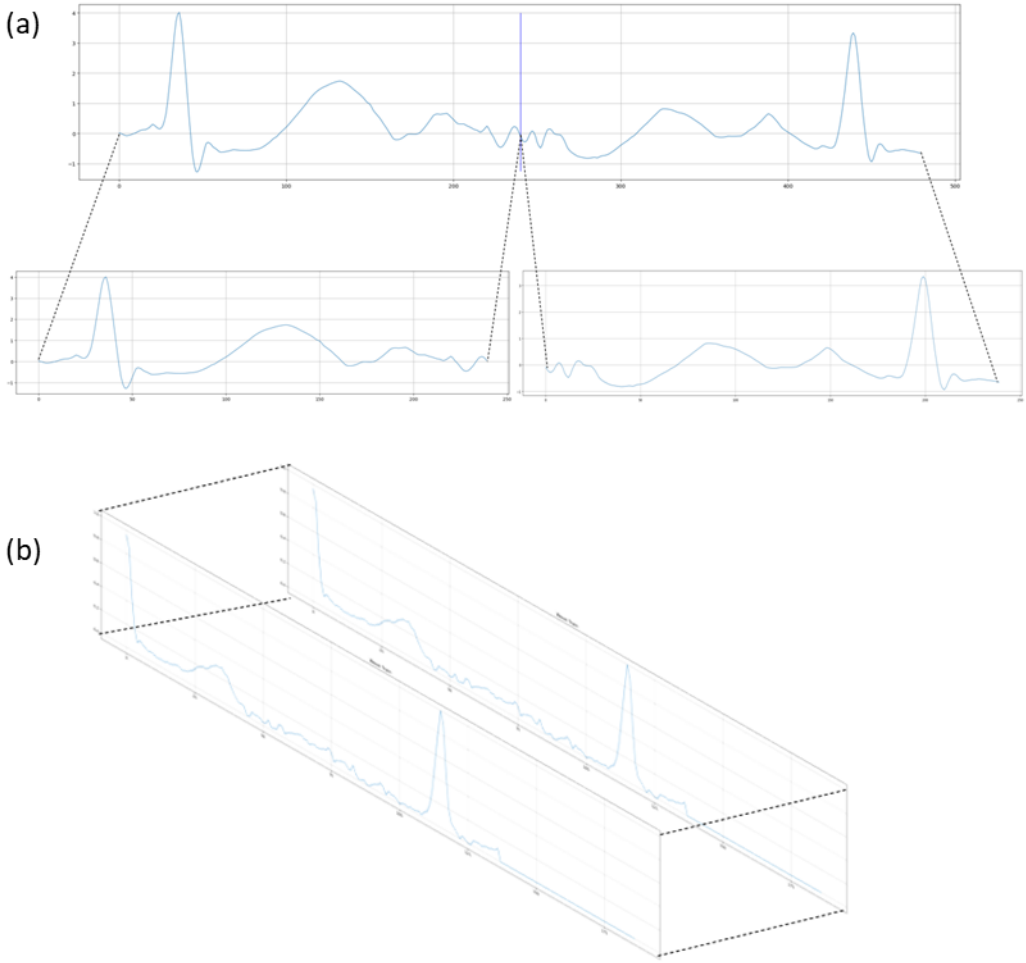


Figure 2: Data Transformation for (a) primary data and (b) MI data

Fig. 2 (a), a full length of 480 beats was broken down into 2 columns of 240 beats each (the blue vertical line represents the split).

For the first validation dataset (VD I), category ‘N’ and its subclasses ‘L’ and ‘R’ were chosen to see if the model can distinguish between similar waveforms. Lesser represented waveforms ‘e’ and ‘j’ were excluded from this analysis as the primary goal was to check if the model can distinguish between closely related waveforms. ‘N’ category samples were also under-sampled to 7000 to ensure class balancing.

For the second MI dataset (validation), the number of data points that were available was around 180. This meant that only one sample was available, and it was not appropriate to further break it down into two channels, unlike the primary dataset. To preserve the network architecture, the input wave was duplicated as shown in Fig. 2 (b).

All the datasets were normalized prior to training. Also, high-class imbalance was observed and as mentioned in the original paper [3], no steps were taken to address it. Rather, the AUC, and ROC values were calculated to evaluate the performance of the entire model and each class.

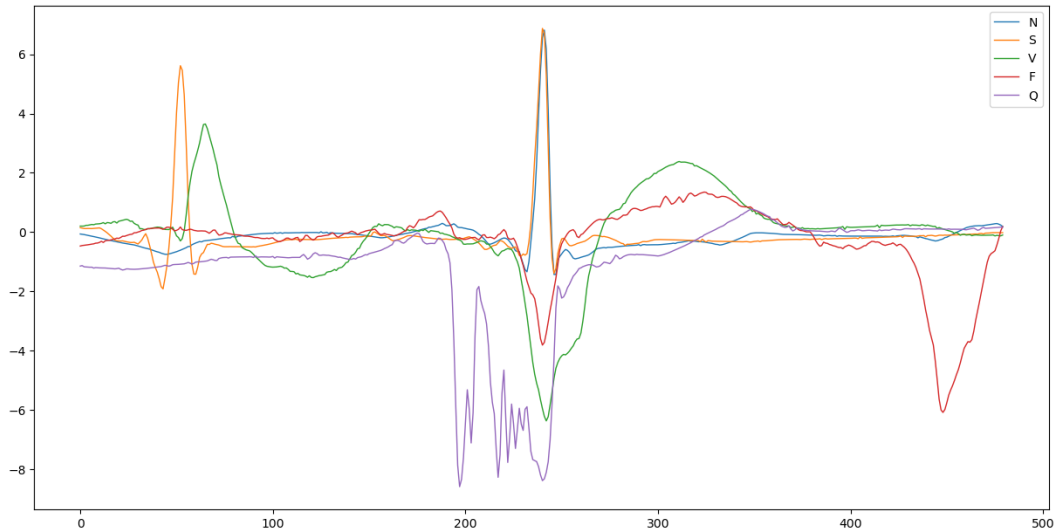


Figure 3: ECG samples belonging to the primary dataset

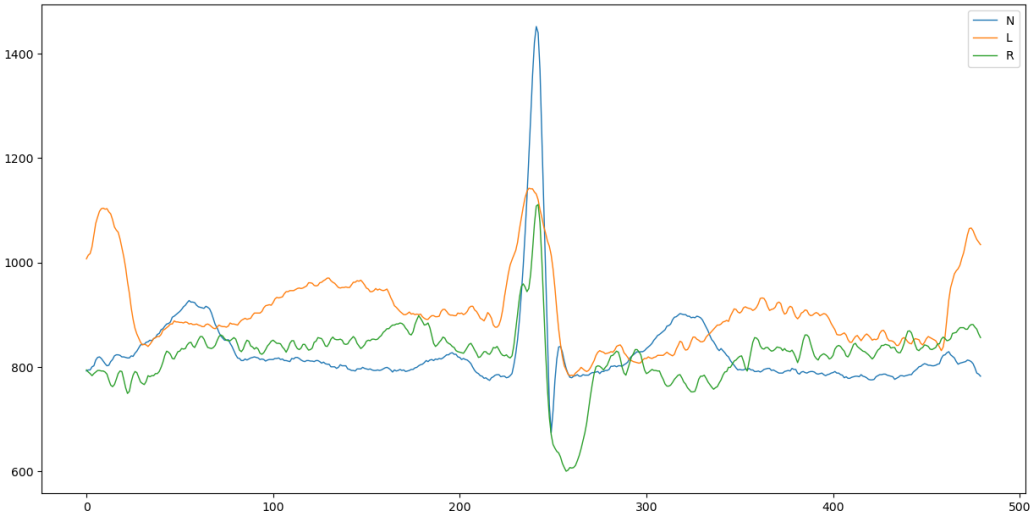


Figure 4: ECG samples representing subclasses of Normal heartbeat

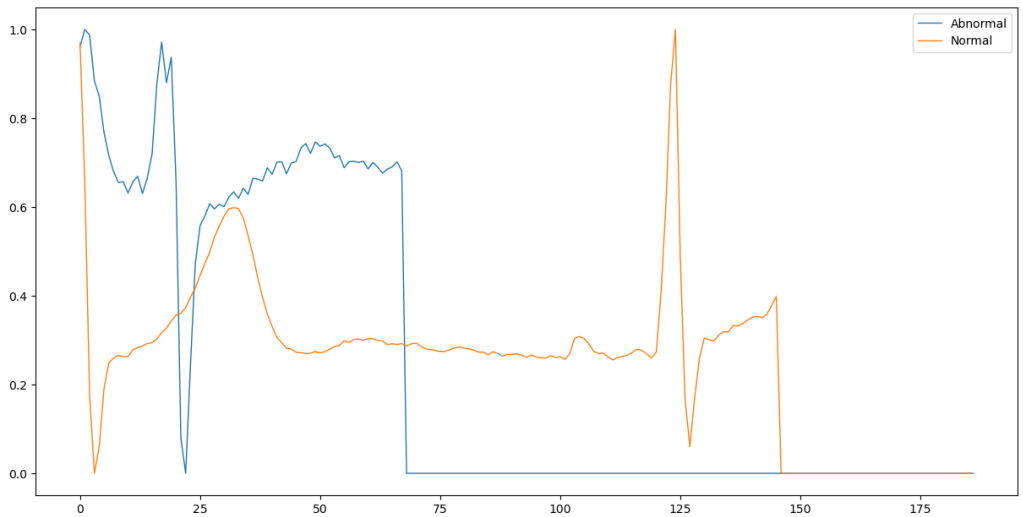


Figure 5: ECG samples representing myocardial infractions

3. Model Training

The architecture that was defined in the original paper [3] was implemented. It was made up of one attention layer, 2 convolution layers, 2 max-pooling layers, and 2 fully connected layers represented in Fig. 6. The training details are as mentioned in Table II. The same training parameters were used for training both the ECG datasets to ensure fair comparison. Furthermore, during learning of the validation dataset the fully connected layer had to be modified to accommodate the change in sample and feature length.

Table II: Hyperparameters for training ABCNN

Parameter	Value
Epochs	50
Learning Rate	0.001
Batch Size	5% of the entire sample
Optimizer	Adam (default parameters)
Loss Function	Cross Entropy Loss
Shuffle	FALSE
Train:Test split	80:20

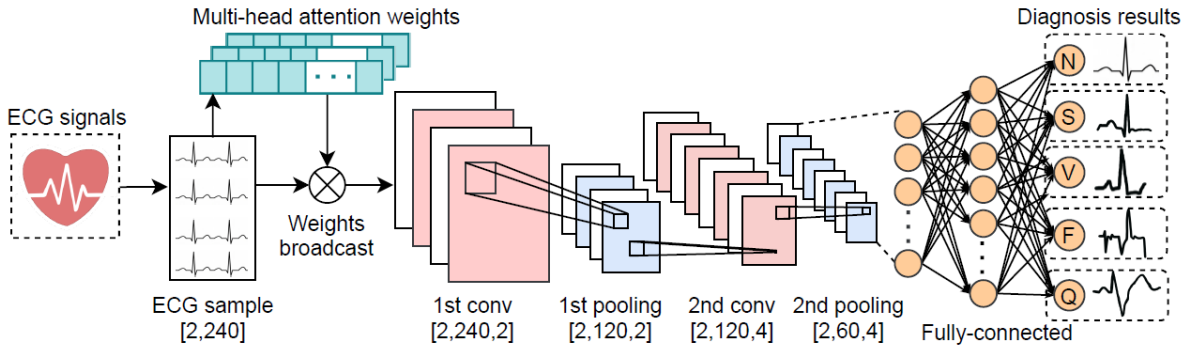


Figure 6: ABCNN representation [3]

4. Model Evaluation

Evaluation metrics such as AUC and ROC were used to understand the performance of the network on the data due to high-class imbalance.

5. Model Fine-Tuning and Transfer Learning

As the dimensions of the primary dataset and validation dataset were different, the fully connected layers had to be modified to accommodate the new dataset. Similarly, transfer learning was also implemented by training on the primary dataset and fine-tuning the model on the validation dataset, after modifying the fully connected layers and freezing the layers that were not altered.

Results

In this section we have summarized the results of the experiments performed. The section is divided into two parts— results pertaining to the primary dataset and the validation dataset.

I. Primary Dataset

The primary dataset consisted of ECG signal data from 48 patients. The data was sampled every 800 milliseconds with 65000 data points. The ECG signal divided into smaller samples of 240 beats per channel with 2 channels as inputs. The final dataset had 96000 samples which was split into training and testing data with ratio 80:20. Fig. 7 shows the distribution of the primary dataset across the different classes. The dataset had 5 classes— Normal beat (N), Supraventricular ectopic beat (S), Ventricular ectopic beat (V), Fusion beat (F), and Unknown beat (Q).

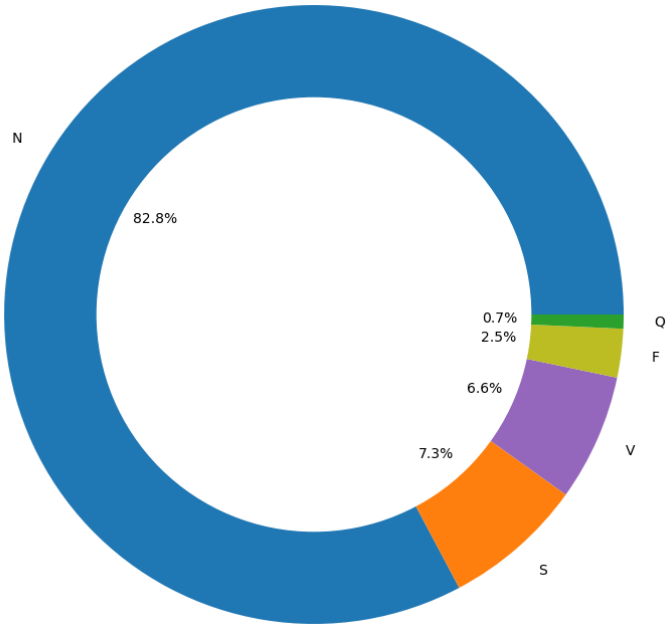


Figure 7: Distribution of classes for primary data

It can be observed from the pie chart that there is class imbalance in the dataset. However, since the paper does not take any corrective action, we proceeded to training the ABCNN model.

The model is trained using the training data which has around 76800 ECG samples. The training spans over 50 epochs, with a *learning rate* of 0.001. *Adam optimizer* is used along with *cross entropy loss*. Fig. 8 show the model loss and accuracy for the training and testing set. It can be observed that the loss curve shows the characteristic decline, indicating convergence. The final accuracy of the model on the testing dataset was 96.85% and the loss was 0.1153.

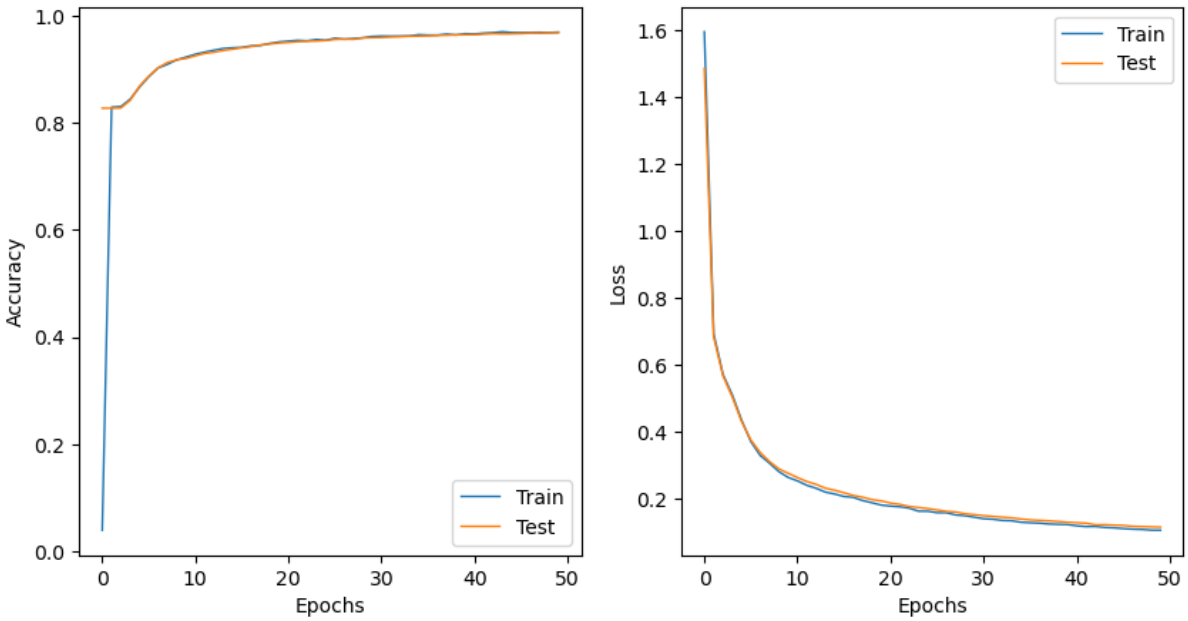


Figure 8: Training and testing accuracy and loss for primary data

Fig. 9 shows the ROC-AUC curve for the different classes. ROC-AUC curve is a method to visualize how well the model can distinguish between positive and negative samples of a class. The 45-degree line represents an AUC of 0.5 which means the model cannot distinguish between the positive and negative samples of a class. The higher the area under the curve (AUC), the better is the model performance. In Fig. 9 the ROC-AUC curves peak close to the left corner of the graph indicating AUC

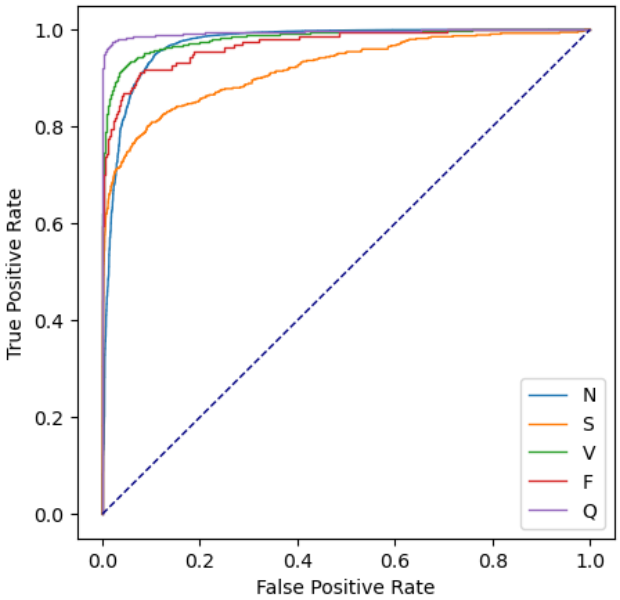


Figure 9: ROC AUC curve for primary data

close to 1. The average AUC for all the classes was observed to be 0.9734.

II. Validation Dataset

Two validation datasets have been used to test the performance of ABCNN. One of the datasets contains samples of the subclasses of a broader category explored in the primary dataset. This allows us to test how the model works on highly complex and intricate data. The dataset had 3 classes— Normal beat (N), Left bundle branch block beat (L), and Right bundle branch block beat (R). The second dataset consisted of signals indicating myocardial infractions. It was a binary classification dataset with Normal and abnormal heartbeats. Fig. 10 shows the distribution of data for the different classes in both the datasets. The normal heartbeat was under-sampled to create a balanced dataset, in order to see if the model performance could be improved.

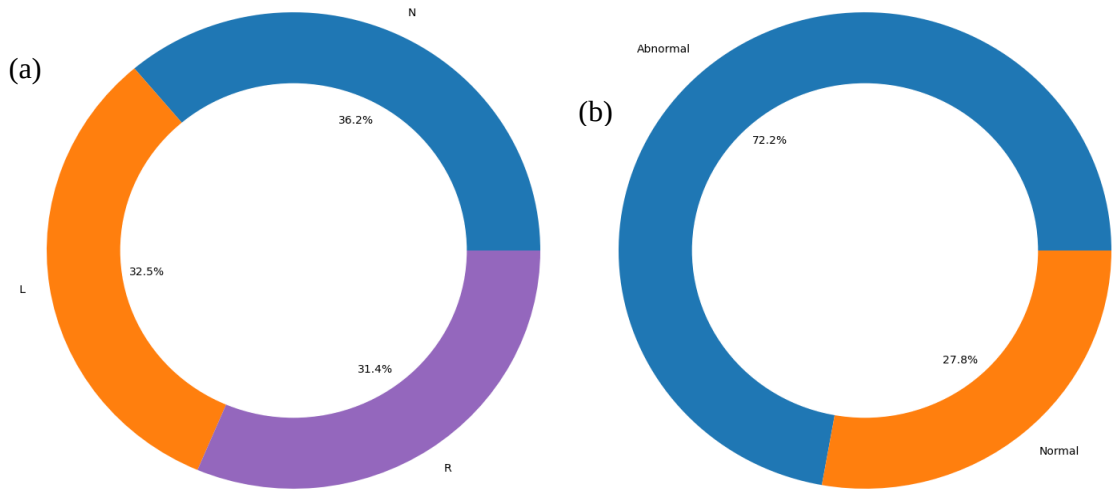


Figure 10: Distribution of classes for (a) VD I and (b) VD II

The final layer was modified to accommodate the different number of classes, but model was trained using the same hyperparameters as for the primary dataset. Fig. 11 represents the training and testing loss and accuracy for both the validation datasets. The trends of the curves indicate that the models converged, and training was successful. The final accuracy and loss for the two validation datasets is reported in Table III. From the table, it can be seen that the performance of ABCNN on the validation data was comparable to the primary data.

Table III: Accuracy and loss for validation data

Dataset	**Training**		**Testing**	
	Accuracy	**Loss**	**Accuracy**	**Loss**
VD I	1	0.0027	0.9940	0.0224
VD II	0.9629	0.1050	0.9533	0.1347

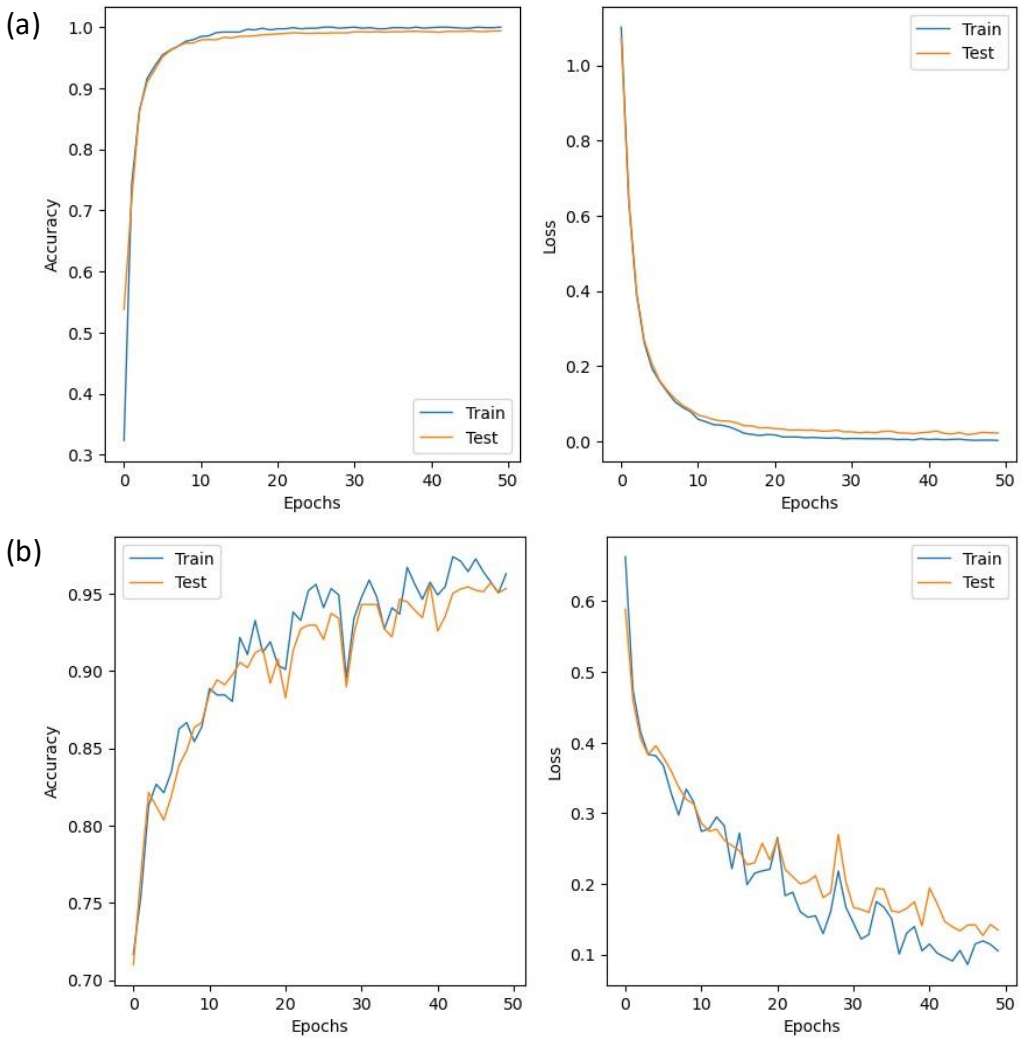


Figure 11: Accuracy and loss plots for (a) subclasses of Normal data and (b) myocardial infraction data

Similar to the primary dataset, we used ROC curves to visualize the performance of the ABCNN on distinguishing between positive and negative samples. The average value of AUC score for the different validation datasets were 0.9983 and 0.9902. From Fig. 12, it can be concluded that ABCNN performs well on both datasets.

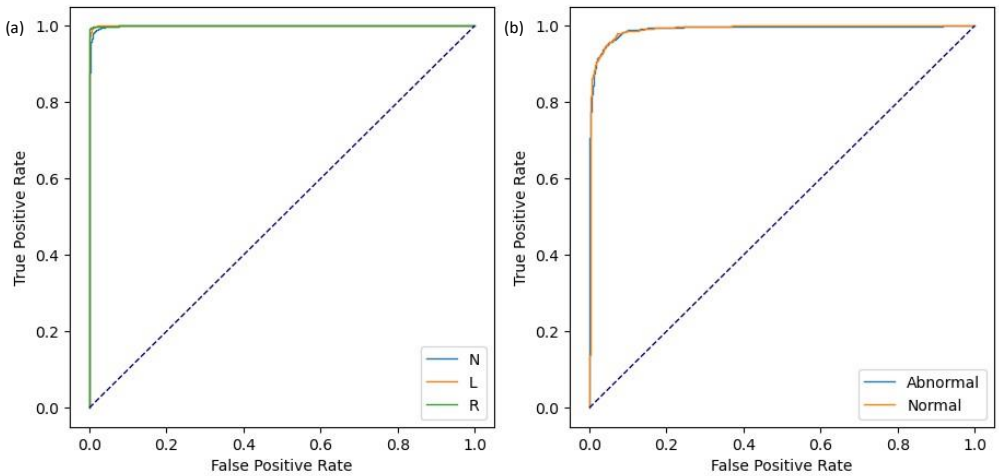


Figure 12: ROC AUC curve for (a) subclasses of Normal data and (b) myocardial infraction data

III. Transfer learning

After training the model on the primary dataset that captured the arrhythmia classes, transfer learning was implemented on VD1 and VD2. The following changes were made to the architecture to facilitate transfer learning on the validation dataset,

- For VD1, the *out* linear layer was modified from $(200, 5)$ to $(200, 3)$.
- For VD2,
    - The *fc* linear layer was modified from $(1984, 200)$ to $(1568, 200)$
    - The *out* linear layer was modified from $(200, 5)$ to $(200, 3)$

The layers that were not modified were frozen and were not included in the optimizer. The AUC score post-training VD1 and VD2 were 0.9502 and 0.9732 respectively. The training and testing loss are as follows,

Table IV: Accuracy and Loss for Validation data post transfer learning

Dataset	**Training**		**Testing**	
	Accuracy	**Loss**	**Accuracy**	**Loss**
VD I	0.8692	0.3466	0.8824	0.3352
VD II	0.9354	0.1715	0.9316	0.1867

The Training and Testing accuracy and loss curves, and AUC curves of TL for VD1 and VD2 are present in the Appendix. These results showed that Transfer learning works well with ABCNN on ECG datasets.

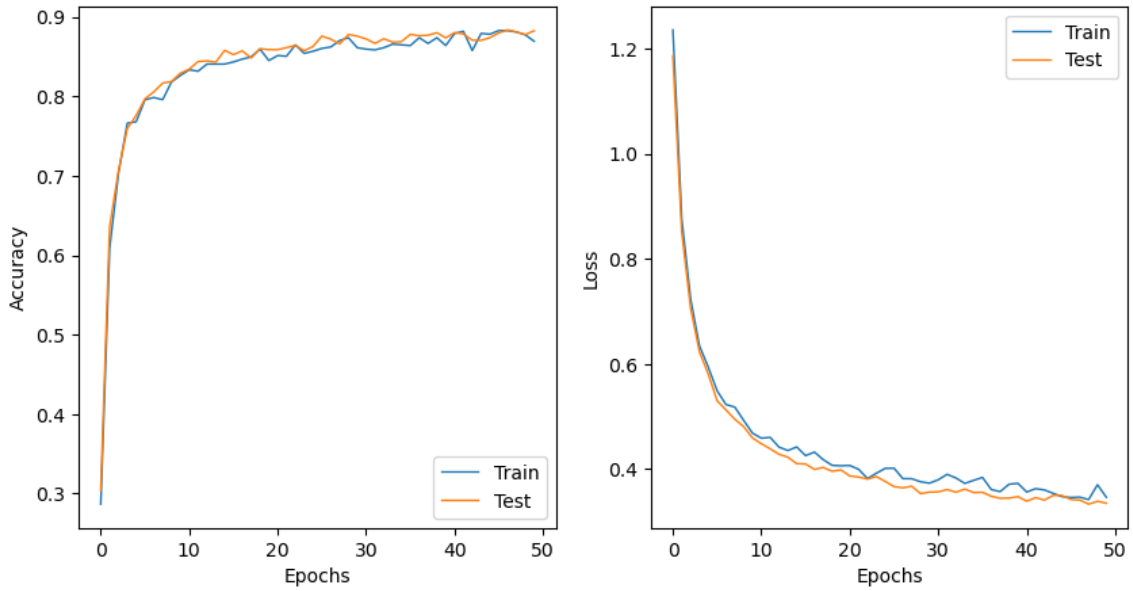
Conclusion

ECG data is a valuable source of information for an individual's health. For obtaining accurate diagnoses, it is imperative that efficient deep learning models are developed. From this study, we concluded that ABCNN omits the need for pre-processing ECG signal, thereby eliminating complexities. The model performed well on different ECG datasets. The validation data was used to check the robustness of the model. The experiments performed revealed that ABCNN produces good results upon training the model from scratch as well as after transfer learning. An average AUC score of 0.9942 and 0.9615 was obtained after training ABCNN from scratch and transfer learning. The superior performance of ABCNN makes it a promising choice for other applications such as text classification and voice recognition.

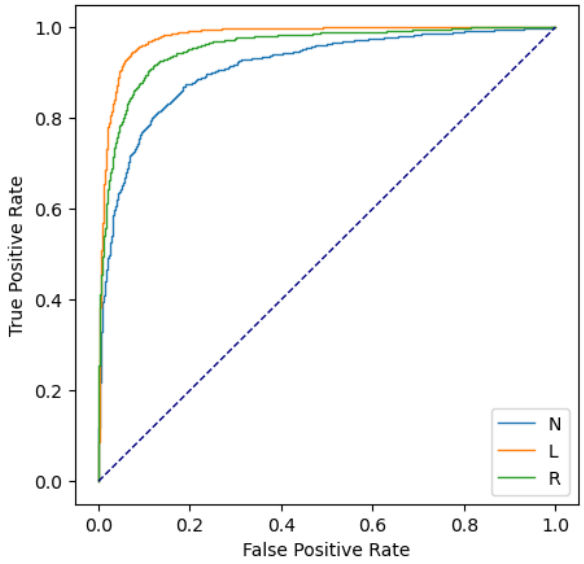
References

[1] Berkaya, S.K., Uysal, A.K., Gunal, E.S., Ergin, S., Gunal, S. and Gulmezoglu, M.B., 2018. A survey on ECG analysis. *Biomedical Signal Processing and Control*, *43*, pp.216-235.
[2] Liu, X., Wang, H., Li, Z. and Qin, L., 2021. Deep learning in ECG diagnosis: A review. *Knowledge-Based Systems*, *227*, p.107187.
[3] Liu, Z. and Zhang, X., 2021, November. ECG-based heart arrhythmia diagnosis through attentional convolutional neural networks. In *2021 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)* (pp. 156-162). IEEE.
[4] Kachuee, M., Fazeli, S. and Sarrafzadeh, M., 2018, June. Ecg heartbeat classification: A deep transferable representation. In *2018 IEEE international conference on healthcare informatics (ICHI)* (pp. 443-444). IEEE.

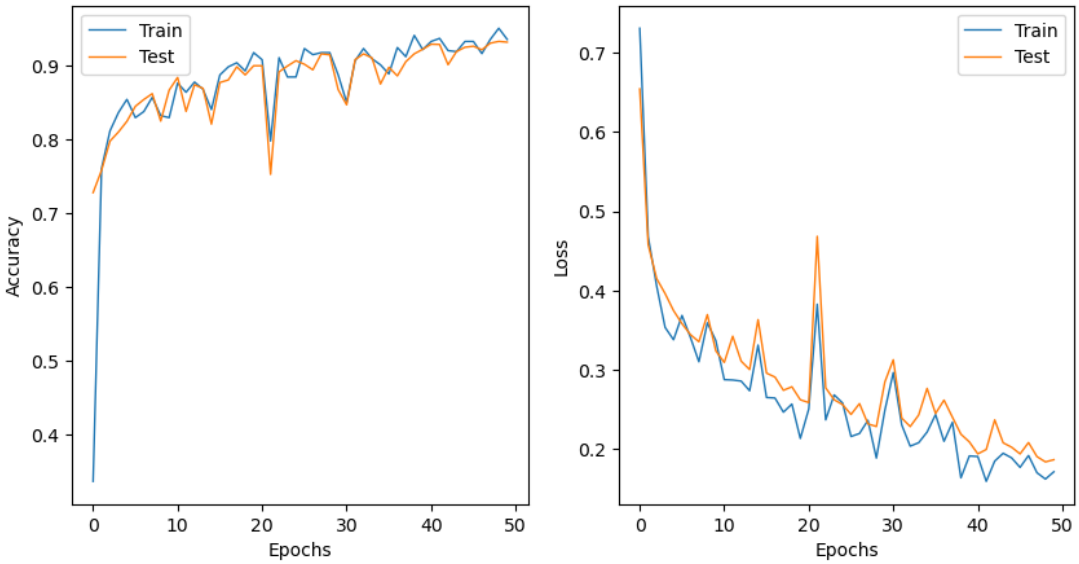
Appendix



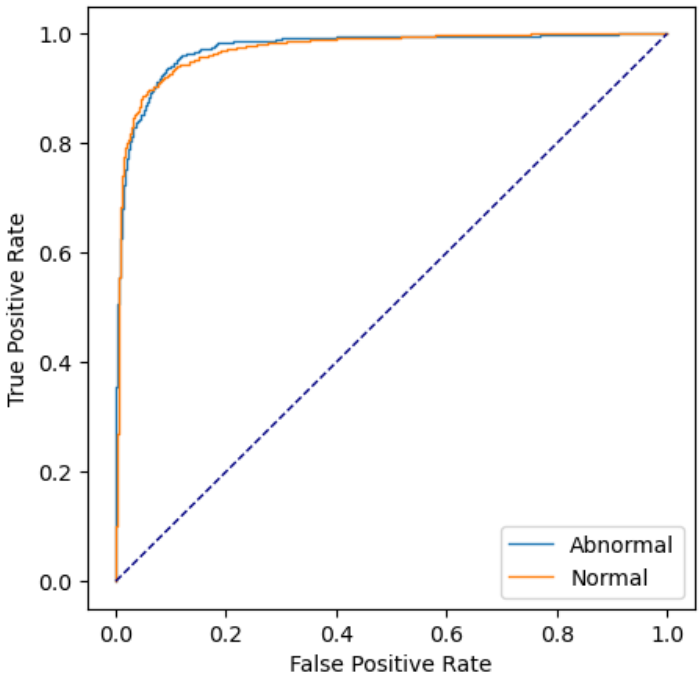
Appendix I.A: Accuracy and Loss curves for VD1



Appendix I.B: AUC for VD1



Appendix II.A: Accuracy and Loss curves for VD2



Appendix II.B: AUC for VD2

Contribution

Riya Saini	1. Implemented original paper with primary dataset. 2. Implemented ABCNN for VD I. 3. Wrote results and conclusions in the report.
Yasomitra Sampat Kota	1. Implemented ABCNN for VD II. 2. Implemented transfer learning. 3. Wrote introduction and methodology.