Comparing Classical Machine Learning and Deep Learning for Classification of Arrhythmia from ECG Signals

Marija Bikova, Vesna Ojleska Latkoska and Hristijan Gjoreski
Faculty of Electrical Engineering and Information Technologies, "Ss. Cyril and Methodius University" in Skopje, Rugjer Boshkovikj 18, Skopje, North Macedonia
marija_bikova@yahoo.com, vojleska@feit.ukim.edu.mk, hristijang@feit.ukim.edu.mk

Keywords: Cardiac Arrhythmia, Deep Learning, Classification Electrocardiogram, Convolutional Neural Network, Long- Short Term Memory.

Abstract: Arrhythmia detection is a vital task for reducing the mortality rate of cardiovascular diseases. Electrocardiogram (ECG) is a simple and inexpensive tool that can provide valuable information about the heart's electrical activity and detect arrhythmias. However, manual analysis of ECG signals can be time-consuming and prone to errors. Therefore, machine learning models have been proposed to automate the process and improve the accuracy and efficiency of arrhythmia detection. In this paper, we compare six machine learning models, namely ADA boosting, Gradient Boost, Random Forest, C-Support Vector (SVC), Convolutional Neural Network (CNN), and Long Short-Term Memory Network (LSTM), for arrhythmia detection using ECG data from the MIT-BIH Arrhythmia Database. We evaluate the performance of the models using various metrics, such as accuracy, precision, recall, and F1-score, on different classes of ECG beats. We also use confusion matrices to visualize the errors made by the models. We find that the CNN model is the best performing model overall, achieving accuracy of 95% and F1-score of 84.75%. SVC and LSTM were the second and third best, achieving accuracy of 94% and 93%, respectively. We also discuss the challenges of using ECG data for arrhythmia detection, such as noise, imbalance, and similarity of classes. We suggest some possible ways to overcome these challenges, such as using more advanced preprocessing and resampling techniques, or incorporating domain knowledge and expert feedback into the models.

1 INTRODUCTION

Cardiovascular diseases (CVDs) have been the leading cause of death since 1999 as the statistics of the Centers for Disease Control and Prevention indicate [1]. The mortality rate can be effectively reduced by providing a timely treatment using a classification model to identify CVDs at early stage [7]. One of the common sources of CVDs is cardiac arrhythmia, where heartbeats are known to deviate from their regular beating pattern. A normal heartbeat varies with age, body size, activity, and emotions. In cases where the heartbeat feels too fast or slow, the condition is known as palpitations. An arrhythmia does not necessarily mean that the heart is beating too fast or slow, it indicates that the heart is following an irregular beating pattern. It could mean that the heart is beating too fast-tachycardia, when there are more than 100 beats per minute (bpm), or slow – bradycardia with less than 60 bpm, skipping a beat, or in extreme cases, cardiac arrest. Some other common types of abnormal heart rhythms include atrial fibrillation, atrial flutter, and ventricular fibrillation [2]. The Electrocardiogram (ECG) signal detects cardiac abnormalities by measuring the electrical signals generated by the heart during contraction. A careful study of ECG signals is crucial for precise diagnoses of patients' acute and chronic heart conditions. Arrhythmia is a cardiac abnormality related to the rate and rhythm of the heartbeat [6]. Despite being the most frequently used diagnosing tool, the rates of ECGs misdiagnosis are still too high. It is very challenging to accurately detect the clinical condition presented by an ECG signal. Cardiologists need to accurately predict and identify the right kind of abnormal heartbeat ECG wave and then recommend the appropriate treatment. The analysis of the electrocardiogram (ECG) signals is done manually which can be time-consuming. To address this issue, machine learning (ML) classification is being

proposed to automate the process. This would allow ML models to learn the features of a heartbeat and detect abnormalities [10][11].

2 ECG STRUCTURE AND MIT-BIH DATABASE

The human body can be thought of as a giant conductor of electrical currents. An electrocardiogram (ECG) can be registered by connecting electrical leads to any two points on the body. The ECG contains records for the electrical activity of the heart. The ECG of the heart forms a series of waves and complexes that have been labelled in alphabetical order: the P wave, the QRS complex, the T wave and the U wave. The P wave is produced by depolarization of the atria; depolarization of the ventricles produces the QRS complex; and repolarization of the ventricles causes the T wave [3]. The significance of the U wave is uncertain. Each of these electrical stimulations results in a mechanical muscle twitch. This is called the electrical excitation-mechanical contraction coupling of the heart. This allows us to detect abnormalities by equating each phase to the normal cardiac cycle. Figure 1 shows the ECG signal representation of a normal beat. These ECG signals are extremely susceptible to high and low frequency noise which usually occur from baseline wander, misplaced electrode contact, motion artifacts, or power line interference [3].

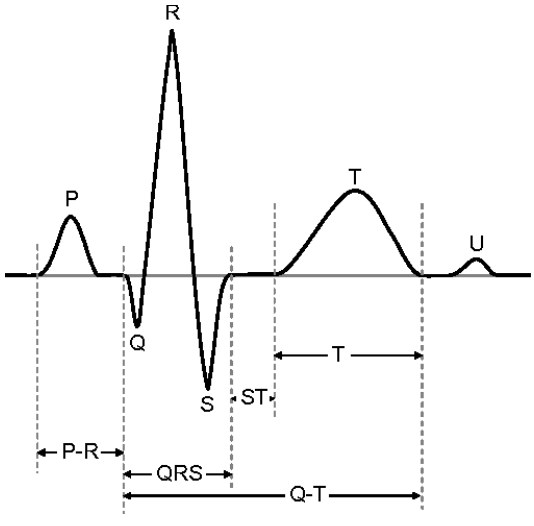

Figure 1: Electrocardiogram (ECG), showing significant intervals and deflections [3].

The MIT-BIH Arrhythmia Database [12] is a publicly available database that contains sections of ambulatory ECG recordings from 47 subjects. The recordings were digitized at 360 samples per second per channel with 11-bit resolution at 10-mV range on two channels and studied by the BIH Laboratory. Here, 23 recordings were picked at random from a set of 4000 24-hour ECG recordings collected from a population of 60% inpatients and 40% outpatients [12]. The dataset has been pre-annotated and labelled by cardiologists. These different annotations refer to various normal and abnormal ECG signals which represent different types of arrhythmia. The dataset consists of ECG signals of various classes, but the eight classes used for this investigation are 'N', 'L', 'R', 'V', 'A', 'F', 'f', and '/'.Table 1 shows the description and numerical identification values assigned to these classes [3].

Table 1: Beat classes, ID number and description.

Class	ID	Beat Description
N	1	Normal
L	2	Left Bundle Branch Block
R	3	Right Bundle Branch Block
V	4	Premature Ventricular Contraction
A	5	Atrial Premature
F	6	Fusion of Ventricular and Normal
f	7	Fusion of Paced and Norma;
/	8	Paced

3 DATA PREPARATION

We used the pre-processing methodology as proposed by Verma et al. [3]. The MIT-BIH dataset was read using the native python waveform-database (WFDB) package, a library of tool for reading, writing, and processing WFDB signals and annotations. Most of the ECG signals were assigned to the annotation classes explained in Table 1. The ECG dataset is imbalanced, since there is an abundance on 'N' beats and the other beat classes do not pass the 10000 thresholds. This is only from one channel of the MIT-BIH database. To get all the beats we extracted and stacked the ECG signals from both the channels. After extracting the 8 classes that are going to be used and removing the other classes, a data clean-up processes were applied. First, the data was made purely numerical in order for easier working, by assigning each of the 8 classes a number, shown in Table 1.

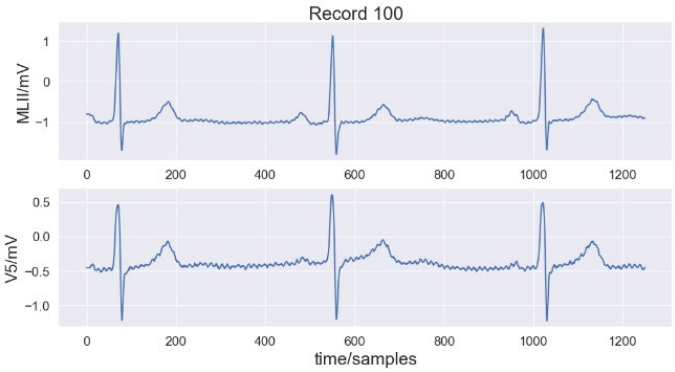

Figure 2: Example of original ECG beats from MIT-BIH database (X-axis: Timestamps, Y-axis: Voltage).

The next step was to make every beat contain equal amount of data points, so each individual beat was extracted from all the records by matching the R-peaks of the ECG with the respective annotation class and appending the class numerical value at the end of the beat. Then standardization process was implemented for all beats to ensure consistent signal amplitudes, using the formula $z=(x-\mu)/\sigma$, where the new beat is represented with z (x is the original beat data, μ is the mean of the beat data and σ is the standard deviation of the data). Each beat was labeled with the patient record number and the annotation class number, and the resulting clean data was then saved into a single .csv file, containing all beats from all records.

The dataset with the clean data was divided into two fundamental subsets, train and test. The test set represented 25% of the original dataset, while the remaining 75% formed the training set.

The original data from MIT-BIH database for one patient is shown on Figure 2. It can be noticed that the ECG signals are continuous and not standardized between the two channels and are sampled at 360 Hz. The data after performing pre-processing is shown in Figure 3, where the standardization and r-peak centering that is implemented on the data points can be seen.

The resample technique by Sci-kit Learn is used in to address the imbalance between the classes in the MIT-BIH dataset. The bootstrap method is involved in this technique of resampling, where statistics are estimated on a data population by sampling a dataset with replacement through iteration using a sample size and number of repeats. By taking the mean values of the total number of beats of the abnormal classes the value for up-sampling and down-sampling denoted as n_samples were calculated. After resampling, all eight classes in the training dataset have 3989 samples for the beat hold out method.

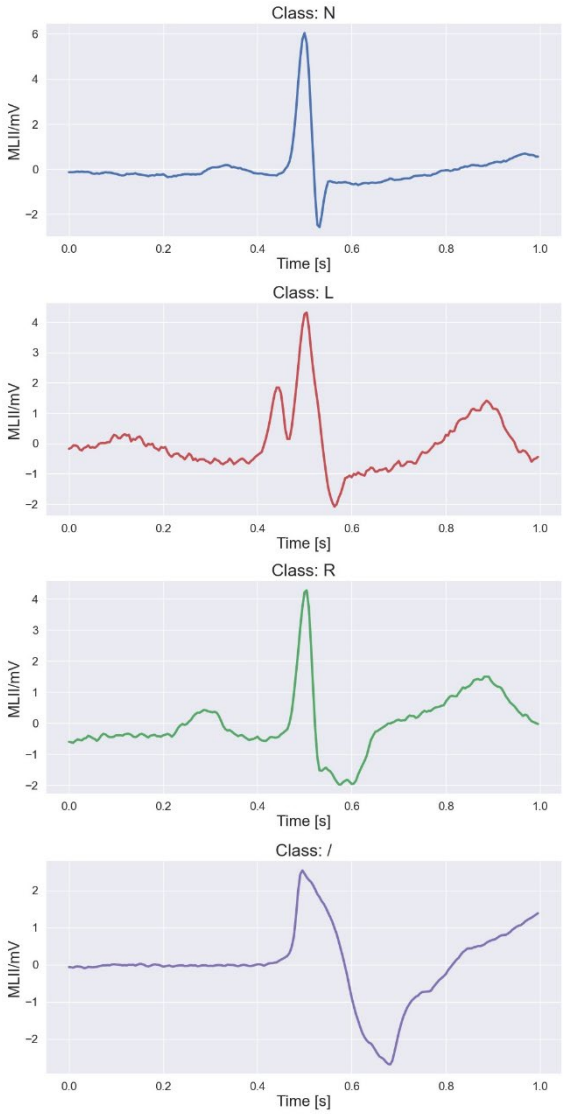

Figure 3: Example of single beats of N, L, R and / classes (X-axis: Timestamps, Y-axis: Voltage).

4 MACHINE LEARNING MODELS

We use 6 models that have different characteristics to classify ECG data into multiple categories based on their patterns. We decided to use deep learning models, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, because of their ability to capture spatial features and temporal dependencies, respectively. This choice was supported by many previous research successes in similar contexts. We also employed classical machine learning models, such as Gradient Boosting (GBC), ADA Boosting (ADA), Random Forecast (RFC), C-Support Vector (SVC), which are well-known for their performance in various classification tasks.

Prior successful research papers [3][4] guided us through the selection of our models. Verma et al. [3], proposed an 11-layer CNN model and LSTM models to classify 8 classes of beats in the MIT-BIH arrhythmia dataset and their models displayed an accuracy of 94.1% and 94% for K-Fold cross-validation method, and 98.7% and 97% for Leave Groups Out method, respectively. Their CNN model had four layers of 1D-convolutiona and batch normalization pairs, with ReLU activation, 16 kernel size and 128, 32 filters. The final two layers were 1D-convolution layer with 9 filters and a 1D-max pooling layer with 2 pool size, followed by a flatten layer and fed to four dense layers. The LSTM model consisted of two LSTM layers with 128 and 9 filters, followed by a 1D-max pooling layer with 2 pool size. The output is flattened and goes to four dense layers with ReLU and softmax activations.

Pandey and Janghel et al. [4] used an 11-layer CNN with SMOTE to classify five beat classes in the MIT-BIH dataset. The network had four 1D-convolution and max pooling layers, followed by two ReLU layers, and a fully connected softmax layer to classify beats into five classes. The model was tested by randomly splitting the beats into training and testing sets, and got 98.3% accuracy.

In this paper, we build on the previous discussion and use the existing CNN and LSTM models as a basis for our design. Our aim is to improve these models and achieve better performance in ECG. classification. We compare different architectures and hyperparameters and evaluate their performance on the MIT-BIH dataset [6]. The accuracy and other metrics for each one of them have been reported and used to identify which models are the best for this problem.

Following on from work discussed in this section, the first CNN model that we proposed for arrhythmia classification is shown in Figure 4. The model consists of two 1D convolutional layers with 64 and 32 filters, with kernel size of 16 and 8 accordingly. Each CNN layer is followed by a batch normalization layer and a max pooling layer with a pool size of 2. The output is flattened and passed through four dense layers with 512, 128, 32, and 9 neurons. The first three dense layers use the ReLU activation and the last layer uses SoftMax activation. We use Adam optimizer, categorical cross entropy loss, and accuracy metric to train the model with a batch size of 64 for 5 epochs.

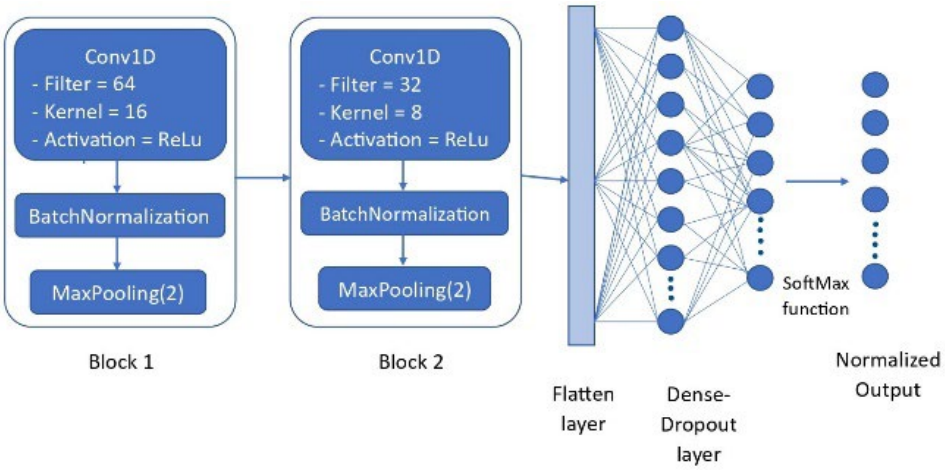

Figure 4: Proposed 1D CNN model architecture.

The second CNN model that we used consists of six 1D convolutional layers. Each convolutional layer applies a one-dimensional filter to the input and uses ReLU activation and 'same' padding. The number of filters increases from 64 to 256 as the layers go deeper, which means that the model can learn more complex and abstract features. Each convolutional layer is followed by a batch normalization layer, which normalizes the output and improves the training speed and stability. After every two convolutional layers, there is a max pooling layer, which reduces the dimensionality of the output by taking the maximum value in each window of size two. There is also a dropout layer, which randomly sets a fraction of input units to zero during training, which helps prevent overfitting. A flatten layer is used to convert the multidimensional sequences into a one-dimensional vector. Then four dense layers are added, the first three dense layers have 512, 128, and 32 neurons, respectively, and use ReLU activation and are followed by batch normalization and dropout layers. The last dense layer has nine neurons and uses SoftMax activation, which is suitable for multi-class classification tasks.

For the third CNN model we used three 1D convolutional layers with 32, 64, and 128 filters, each followed by batch normalization and max pooling layers with pool size of 2. The output is flattened and passed through dense layers with 256, 64, and 9 neurons. After the first two dense layers there is a dropout layer with a rate of 0.2 and they use ReLU activation. The last dense layer uses SoftMax activation.

All the CNN models that we experimented with in this paper have the same training settings. They use Adam optimizer, categorical cross entropy loss, and the models are trained with a batch size of 256 for 5 epochs [8].

The first LSTM model proposed and investigated in this paper uses the Keras Sequential API and consists of 8 weighted layers. The first layer has 128 units and returns sequences. After the first LSTM layer a dropout regularization with a rate of 0.2 is applied. Dropout randomly sets a fraction of input units to 0 during training, which helps prevent overfitting. The second LSTM layer has 64 units and after this layer a dropout regularization with a rate of 0.2 is applied. Next to reduce the temporal dimensions of the sequences a Max-Pooling layer with a pool size of 2 is applied. Then a flatten layer is used to convert the multidimensional sequences into a one-dimensional vector, preparing the data for the fully connected layers. The model uses three dense layers, the first dense layer has 256 neurons and uses the ReLU activation function, the second dense layer has 128 neurons and uses the same activation function ReLU. After each of these two dense layers a dropout regularization with a rate of 0.3 is applied. The final dense layer has 9 neurons with the SoftMax activation function, suitable for multi-class classification tasks.

The second LSTM model has two LSTM layers with 256 and 128 units, each followed by a dropout layer with a rate of 0.3. Then a dense layer with 64 neurons with a ReLU activation is applied, followed by a dropout layer with a rate of 0.3 and another dense layer with 9 neurons with SoftMax activation function.

The third LSTM model is similar to the second, but instead of two LSTM layers it has three LSTM layers with 512, 256 and 128 units. We increased the dropout rates for each dropout layer in the third model to 0.5 which are higher than the 0.3 of the second model, to increase the robustness and generalization ability of the model and also to prevent overfitting.

All LSTM models use the Adam optimized, with the default learning rate of 0.001, categorical cross-entropy loss and accuracy metric. The models are fit on the training dataset for 5 epochs and 256 batch size [9].

For the remaining models, the same hyper-parameters as in [3] were used: GBC and ADA: n_estimators = 100; RFC: n_estimator = 10 and max_depth = 10 ; SVC: default parameters.

The proposed models were trained on a HP notebook equipped with an Intel Core i5-7200U with 2 cores, running at 2.50GHz (2.71GHz turbo boost). 8GB internal RAM and 1TB internal SSD hard drive. Substantial computational power and training time were needed to train the CNN and LSTM models. The SSD storage enabled fast data access, and the setup was affordable. While the absence of the GPU made the training times longer, but still the project was completed successfully on this hardware setup.

5 RESULTS

A comprehensive representation of the performance metrics of the ten classification models used in this paper is provided in Figure 5. Specifically, it displays the accuracy and standard deviation for each of these models, allowing for a visual comparison of their predictive capabilities and consistency. These results indicate that CNN is the best model for arrhythmia detection, as it correctly classified most of the ECG beats in all classes and made fewer errors than the other models. However, LSTM and SVC also performed well and may have some advantages over

CNN in terms of computational complexity and robustness, respectively.

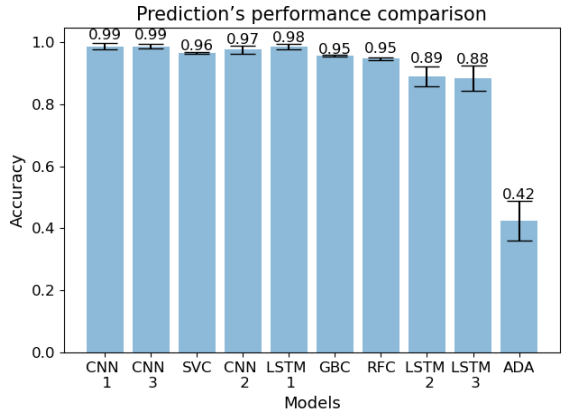

Figure 5: Achieved accuracy for each of the 10 models.

To measure and compare the performance of the ML models and estimate its general performance, cross-validation is used. The data is split into 75% of training set and 25% of test set. The accuracy and other metrics are reported on the test set as the performance of the classifiers. To have a clear view of the model's performance on each class, we measured three metrics: precision, recall, and F1-score, for each of the classes. The results also show the weighted average of these metrics across all classes, taking into account the imbalance of the classes. Due to conciseness and in order to highlight the most relevant findings, in Figure 6 we will showcase results for the top three best-performing models only.

The classification results show that the three CNN models had the highest accuracy of around 95% to 96%, followed by the SVC model with 94% and the first LSTM model with 93% accuracy. The first graph in Figure 6 illustrates the results achieved using the first CNN classifier. It is obvious that the CNN model succeeded in predicting all the classes with different performances. The most predictable classes for CNN are the N and '/' classes, the highest recall (100%) is achieved for the L class and F1-score (99%) for the R class. The second graph show the results achieved by the SVC model, which has similar results as the CNN model, the highest recall (99%) and F1-score (99%) for the L and R classes. The best LSTM model of the three LSTM models had the highest recall (100%) and F1-score (99%) for the L class. The other models achieved significantly lower results, i.e., GBC: 91%, RFC: 90%, ADA: 26%. This shows that the ADA classifier significantly underperforms with the default hyperparameters, requires hyperparameter tunning.

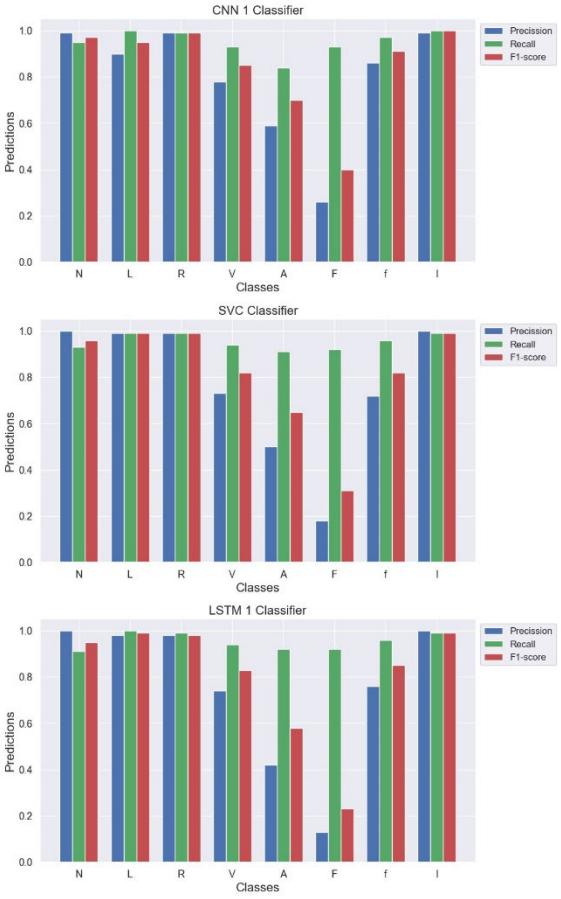

Figure 6: Graphical representation of the classification report for the top 3 models according to accuracy.

Figure 7 illustrates the confusion matrices for the three best models, the diagonal of these confusion matrices shows that most of the models have more than 80% accuracy in classifying the ECG beats. The L, R and / classes are the easiest to classify, with over 90% accuracy for all the models. The N class is the hardest to classify, especially for ADA, which only has about 20% accuracy. The other models have over 80% accuracy for the N class. ADA is very sensitive to noise and outliers, and it learns gradually. Therefore, ADA is not suitable for ECG beat classification. It can be noticed that classes A and F are the most difficult to classify correctly. This might be explained by the fact that class A beats have a similar shape to class N beats, which makes them hard to distinguish. On the other hand, class F beats are very uncommon in the dataset, and the simple method of up-sampling them does not help the models to learn their features well enough.

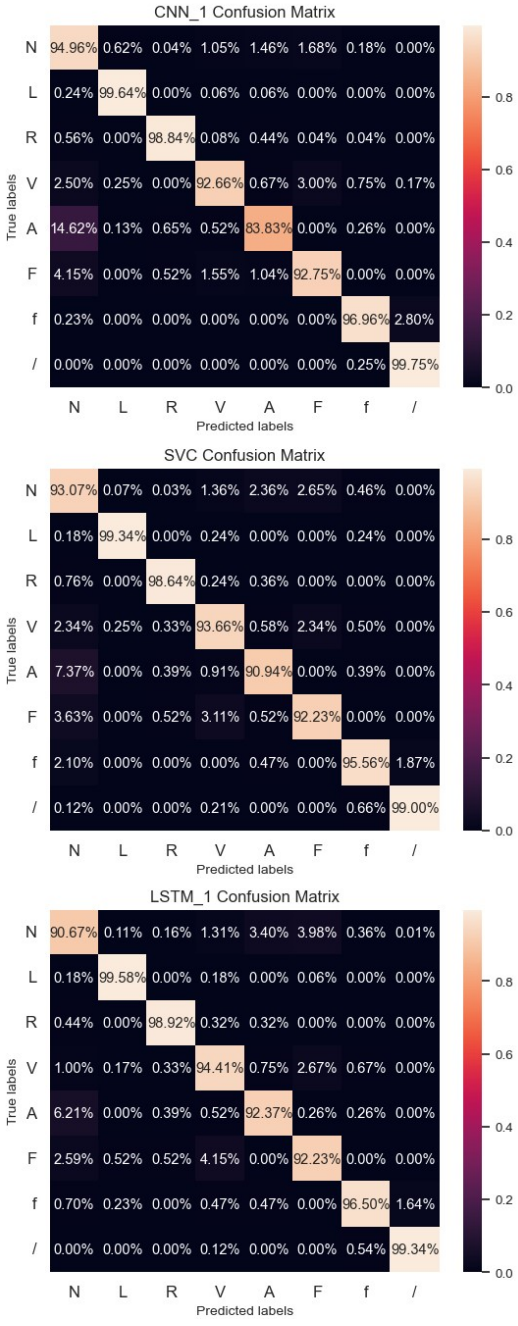

Figure 7: Confusion matrices for the top 3 models according to accuracy.

6 CONCLUSIONS

The paper presented a thorough comparison of 2 Deep Learning approaches (CNN and LSTM) to 4 classical Machine Learning models (GDC, ADA, RFC, SVC) to classify arrythmia from ECG data. In total we used 10 models (3 for each DL approach and 1 for each ML approach) and compared their performance on the MIT-BIH Arrhythmia database. Various metrics such as accuracy, precision, recall and F1-score we used to evaluate the models on different classes od ECG beats. The results showed that CNN is the best model overall for arrhythmia detection, achieving the highest accuracy of 95% and the highest F1-score for most of the classes. SVC and LSTM also performed well, with accuracy of 94% and 93%, respectively, and high F1-scores for some classes. However, LSTM and SVC may have some advantages over CNN in terms of computational complexity and robustness, respectively. The ECG data can present several challenges for the arrhythmia detection models, such as noise, imbalance, and similarity of classes. To overcome these challenges, more advanced preprocessing and resampling techniques should be used. Incorporating domain knowledge and expert feedback into the models is a promising direction for future research and development. We believe that with further research on improving the performance and interpretability of machine learning models for ECG data analysis, more precise results can be expected.

REFERENCES

[1] K.Mc Namara, H. Alzubaidi, and J.K. Jackson, “Cardiovascular disease as a leading cause of death: how are pharmacists getting involved?” Integrated Pharmacy Research and Practice, vol. Volume 8, no. 8, pp. 1-11, Feb. 2019, [Online]. Available: https://doi.org/10.2147/iprp.s133088.

[2] A. Ullah, S.M. Anwar, M. Bilal, and R.M. Mehmood, “Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation,” Remote Sensing, vol. 12, no. 10, p. 1685, May 2020, [Online]. Available: https://doi.org/10.3390/rs12101685.

[3] S. Verma, “Development of Interpretable Machine Learning Models to Detect Arrhythmia based on ECG Data,” May 2022, [Online]. Available: https://doi.org/10.48550/arxiv.2205.02803.

[4] S.K. Pandey and R.R. Janghel, “Automatic detection of arrhythmia from imbalanced ECG database using CNN model with SMOTE,” Australasian Physical & Engineering Sciences in Medicine, vol. 42, no. 4, pp. 1129-1139, Nov. 2019, [Online]. Available: https://doi.org/10.1007/s13246-019-00815-9.

[5] M.T. Le, V.S. Rathour, Q.S. Truong, Q. Mai, P. Brijesh, and N. Le, “Multi-module Recurrent Convolutional Neural Network with Transformer Encoder for ECG Arrhythmia Classification,” Jul. 2021, [Online]. Available: https://doi.org/10.1109/bhi50953.2021.9508527.

[6] S. Armstrong, "Survey of Machine Learning Techniques To Predict Heartbeat Arrhythmias," arXiv.org, Aug. 22, 2022, [Online]. Available: https://arxiv.org/abs/2208.10463, [Accessed on Nov. 04, 2023].

[7] J. Wu et al., "Editor's Choice - Impact of initial hospital diagnosis on mortality for acute myocardial infarction: A national cohort study," European Heart Journal: Acute Cardiovascular Care, vol. 7, no. 2, pp. 139-148, Aug. 2016, [Online]. Available: https://doi.org/10.1177/2048872616661693.

[8] S.K. Pandey and R.R. Janghel, "Automatic detection of arrhythmia from imbalanced ECG database using CNN model with SMOTE," Australasian Physical & Engineering Sciences in Medicine, vol. 42 (4), pp. 1129-1139, November 2019.

[9] J. Gao, H. Zhang, P. Lu, and Z. Wang, "An Effective LSTM Recurrent Network to Detect Arrhythmia on Imbalanced ECG Dataset," Journal of Healthcare Engineering, October 13, 2019.

[10] A. Mustaqeem, S.M. Anwar, and M. Majid, "Multiclass classification of cardiac arrhythmia using improved feature selection and SVM invariants," Computational and Mathematical Methods in Medicine, vol. 2018, pp. 1-10, 2018, doi: 10.1155/2018/7310496.

[11] A. Mustaqeem, S.M. Anwar, M. Majid, and A.R. Khan, "Wrapper method for feature selection to classify cardiac arrhythmia," IEEE Xplore, Jul. 01, 2017, [Online]. Available: https://ieeexplore.ieee.org/document/8037650, [Accessed on Jan. 11, 2023].

[12] G.B. Moody and R.G. Mark, "The impact of the MIT-BIH Arrhythmia Database," IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45-50, 2001, [Online]. Available: https://doi.org/10.1109/51.932724.