Evaluation of Different ML Models for Automated Atrial Fibrillation Detection on ECG Data - Group 2

Danial Beg
29043292
dbeg@uci.edu

Sahithi Chimmula
53443964
schimmul@uci.edu

1 Introduction

The objective of this research was to develop a diagnostic tool for the identification of irregular cardiac activity, with a particular focus on atrial fibrillation (AFib), by analyzing electrocardiogram (ECG) waveform data. This study utilizes a subset of the PTB-XL dataset (Bousseljot et al., 2020), which is an extensively accessible repository of electrocardiographic (ECG) information, to discern whether a patient exhibits a normal sinus rhythm, arrhythmia, or AFib. To achieve this, the project explores the application of several machine learning models, including a transformer architecture (encoder + classification layer), a Long Short-Term Memory network (LSTM), a Multi-Layer Perceptron (MLP), and logistic regression. The F-1 precision, and recall scores will be used to evaluate model performance.

Atrial fibrillation (AFib) constitutes a significant and escalating challenge within healthcare, characterized by irregular and rapid heart rhythms. The mortality associated with AFib as a primary or contributory cause has witnessed a marked increase over the past two decades (CDC, 2023). Early detection and diagnosis of AFib is crucial, as timely intervention can reduce the risk of stroke by 66% (Get, 2023), making any advancement in this space one that can have great benefits for potential patients. The adoption of machine learning (ML) in clinical settings promises to enhance the identification of patients with irregular cardiac rhythms, thus improving classification accuracy and facilitating the early detection of AFib cases, with the potential to save lives. The inputs for this will consist of ECG data sourced from 12 distinct electrodes affixed to the body, capturing cardiac activity. This data will be processed through various models under investigation in this project, yielding outputs that categorize heart rate activity as normal, arrhythmic, or AFib. All code for this project can be found on GitHub.

2 Related Works

Previous research in this domain has investigated the use of individual models, such as a convolutional neural network (CNN), which achieved an F-1 score of 88.2% and an accuracy rate of 97.3% (Wei et al., 2022). Further studies have engaged with transformers that utilize component awareness, deconstructing ECG waveforms into discrete components and encoding them as singular vectors characterized by length and type (Yang et al., 2022). This project seeks to synthesize the methodologies employed in these disparate studies, applying a multitude of models to a uniform dataset and employing consistent evaluation metrics to ascertain the most effective model for detecting cardiac irregularities.

3 Data Sets

The data set we want to employ in this study, as mentioned above, takes data from multiple different electrodes placed throughout the body (from the chest to different limbs as well) and plots a sinus rhythm of a patient's heart activity. The input data is already pre-processed and flattened into values stored into an array that track cardiac activity amongst the different recording electrodes. The dataset includes additional information that can provide value regarding a patient such as weight, height, sex, and age. There exist three different labels in the dataset as well for heart rhythm: 0 for normal heartbeat, 1 for AFib, and 2 for arrhythmia.

The data presented in Table 1 illustrates the distribution of labels in the dataset, highlighting a relative balance among the different categories. Specifically, the table shows that the category of other arrhythmia (VA) holds the highest count with 2841 instances, indicating a significant presence in

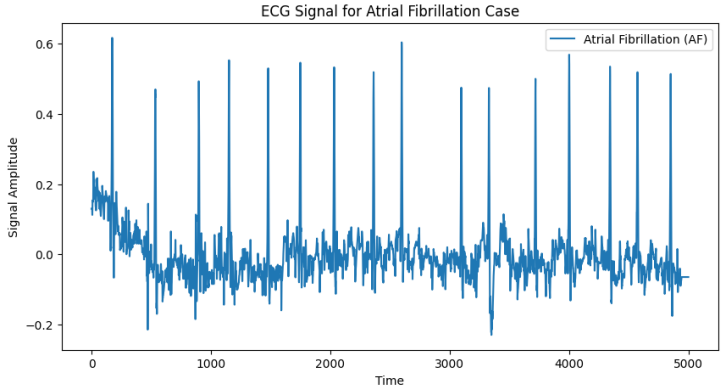

Figure 1: Atrial Fibrillation ECG Waveform Data

Label	Number of Instances
Normal (SR)	2000
Atrial Fibrillation (AF)	1587
Other arrhythmia (VA)	2841

Table 1: Distribution of Labels in Dataset

the dataset. In contrast, Atrial Fibrillation (AFib) has the fewest occurrences, with 1587 instances recorded. Normal cases, labeled as Sinus Rhythm (SR), represent a substantial portion as well, with a total of 2000 instances. This distribution suggests a diverse dataset, with a notable emphasis on other types of arrhythmia.

Figure 1 gives an example look of some ECG waveform data of a patient with AFib. Unlike in normal sinus rhythm, AFib is characterized by irregular R-R intervals, which are the distances between consecutive R waves (the peaks corresponding to the ventricular contractions in the figure). In a standard ECG, regular R-R intervals follow a relatively predictable pattern, whereas in AFib, these intervals vary unpredictably. This phenomena can be observed in the figure as we see an irregular sequence of R-R intervals, indicating the absence of a consistent heart rhythm. Additionally, the ECG baseline shows irregular oscillations instead of smooth, regular waves (known as P waves), which is another hallmark of AFib.

Looking at the additional patient data in Table 2, we can see the information we utilized in the training of our models from patient age to sex, height, weight along with other metrics. These help give a better, more holistic view of a patient's demographics - helping lead to an overall more effective model.

Column Name	Data Type
I	float64
II	float64
III	float64
aVF	float64
aVR	float64
aVL	float64
V1	float64
V2	float64
V3	float64
V4	float64
V5	float64
V6	float64
age	float64
sex	float64
height	float64
weight	float64
nurse	float64
site	float64
device	float64

Table 2: DataFrame Columns and their Data Types

4 Technical Approach

The dataset has been pre-processed using signal processing techniques and it comprises of 12 ECG leads distributed across the body, along with demographic variables such as weight, height, sex, and age. To begin, we will undertake exploratory data analysis (EDA) to aid in feature engineering. This process is crucial for uncovering various patterns and relationships among the features, providing insights that are essential for the effective engineering of features. Next, the dataset will be divided into a training (75%) and test(25%) split in order to facilitate model training and evaluation. This initial phase involves training

and evaluating logistic regression and MLP models as baseline benchmarks. These models, lacking sequential capabilities, serve as the foundation before exploring more advanced options like LSTM and Transformer models. For these models, we will perform hyperparmeter tuning such as varying the number of epochs or iterations, learning rate, batch size, and regularization parameter. The second phase of the project involves harnessing models adept at processing sequential data and capturing long-term dependencies. Each patient's ECG data comprises 700 recorded time points for each of the 12 ECG leads. For logistic regression and MLP, this matrix is flattened, treating each time point independently. In contrast, for LSTM and Transformer models, the temporal sequences require grouping all time points for each patient to comprehend the inherent sequence information. In continuation of the machine learning pipeline from the first part, the second phase will introduce additional hyperparameters, including the number of encoder layers and LSTM units. Further exploration will involve the consideration of various positional encoding methods and weight initialization techniques if deemed necessary. We choose to evaluate our model using F-1 score (trade-off between false positives and false negatives), recall (TP/ FN + TP) and precision (TP/TP + FP) as this is a medical problem where we are more concerned about correctly identifying positive cases. We will also track the loss and accuracy of the models as they train over multiple epochs to get a gauge on how the model trains.

5 Software

We primarily utilized Python as the primary programming language for our project. Our approach involves leveraging libraries such as PyTorch for model development, alongside traditional data science libraries such as Pandas, NumPy, and SciKit-Learn—for data manipulation and visualization. Google Colab served as our platform to host and collaboratively work on the notebooks. Our objective included building a transformer model from scratch and using the aforementioned libraries to prepare data for integration with other models. We also developed code to analyze data through graphical representations and compare F-1 accuracy scores along with precision and recall. Additionally, we conducted hyperparameter tuning to optimize our results. Given the complexity of our training data, we also utilized Google Colab Pro in order to employ their A100 GPUs with high RAM allocations.

6 Experiments and Evaluation

6.1 Methods

The experiments carried out first delved into implementing the aforementioned models (logistic regression, MLP, LSTM, and a classical transformer) and then performed hyper-parameter tuning to improve the accuracy of the model on the data, and finally finishing with evaluating the results and comparing it between the models. Overall, the experiments all ran on a train-test split of 75%-25% for the logistic regression, MLP, and LSTM, and a split of 90%-10% for the transformer with the ritmi label of regular heartbeat, AFib, or arrhythmia being the Y and the X being the dataframe as outlined by Figure 2. We evaluate these models utilizing F-1 score as a metric which is the trade-off between false positives and false negatives. F-1 score is important for our uses as it takes into account both precision and recall and provides a good real-world metric of the robustness of a classification model. Additionally we plan to utilize precision and recall where precision measures the accuracy of positive predictions thus placing an importance of the number of false-positives which would be important in this use case to avoid giving false diagnoses. Recall would measure the ability to find all relevant cases in the data and is a very important metric to measure how many patients are misdiagnosed as not having an irregular heart beat, when they actually do which would be a large issue.

6.2 Logistic Regression

We first explored a basic logistic regression model as a baseline. We will utilize logistic regression as a benchmark in order to judge the accuracy of F-1 scores of the other implemented models, aiming to beat this model in order to show that adding complexity in terms of the model yields better results and captures the data better. For logistic regression, we tuned the hyperparameter - C - in order to achieve the highest accuracy and corresponding F-1 score. As shown in Table 4, logistic regression had an accuracy of 47.92%. Although this accuracy isn't very high, it still

Model	Label	Precision	Recall
Logistic Regression	Regular	0.46	0.15
Logistic Regression	AFib	0.44	0.49
Logistic Regression	Arrhythmia	0.50	0.71
MLP	Regular	0.81	0.65
MLP	AFib	0.71	0.77
MLP	Arrhythmia	0.74	0.80
LSTM	Regular	0.90	0.86
LSTM	AFib	0.82	0.93
LSTM	Arrhythmia	0.92	0.88
Transformer (Balanced Loss)	Regular	0.35	0.94
Transformer (Balanced Loss)	AFib	0.17	0.02
Transformer (Balanced Loss)	Arrythmia	0.21	0.01

Table 3: Precision and Recall for Logistic Regression, MLP, and the LSTM

Model	Accuracy
Logistic Regression	47.92%
MLP	75.34%
LSTM	93.04%
Transformer	46.57%
Transformer (Balanced Loss)	33.40%

Table 4: Tested models with their corresponding best accuracy achieved

Model	Regular	AFib	Arrhythmia
Logistic Regression	0.22	0.59	0.46
MLP	0.72	0.77	0.74
LSTM	0.89	0.87	0.90
Transformer (Balanced Loss)	0.50	0.04	0.03

Table 5: Tested models with their corresponding best F-1 score achieved

performs better than random chance given that we have 3 labels. Additionally, the highest F-1 score for a single label for this model was 0.59 for the AFib label and the lowest F-1 score being 0.22 for regular as shown in Table 5. These F-1 scores suggest the model is performing at best at a moderate level on the dataset for AFib and arrhythmia with it having a relatively balanced performance for detecting AFib. However, the performance for regular heartbeats is quite bad and especially in the context of healthcare where a false negative can have potentially devastating effects, this is not acceptable.

The model shows a limited ability to correctly identify regular heartbeats, with a precision of 0.47 and a recall of 0.15 as shown in Table 3. This suggests that while almost half of the instances classified as regular heartbeats are correct, the model fails to identify a significant number of actual regular cases. For the AFib label, precision is slightly lower than for the regular label at 0.44, which shows that a smaller proportion of AFib predictions were correct. However, recall is higher at 0.49, which means the model is better at identifying AFib cases among the actual AFib instances, but still misses more than half. This model performs best in identifying arrhythmias with a precision of 0.50 and the highest recall of 0.71 among the three labels. This indicates that the model is more reliable in detecting arrhythmia cases than regular or AFib cases. This may be due to the larger number of labels for arrhythmia cases which leads to a higher recall and precision.

Regardless, logistic regression will serve as a good benchmark to work with for the following models, aiming to beat this and have an F-1 score above 0.5 for the other models to have an at least balanced performance between the precision and recall and ideally working towards good performance: an F-1 score above 0.7. These scores will also hopefully follow with higher precision and recall values above 0.7 respectively.

6.3 MLP

The next model deployed for this problem space is a Multi-Layer Perceptron (MLP). The MLP deployed consists of an input layer, two hidden layers, and an output layer. The forward function defines how the input data flows through the network: it first passes through the first hidden layer which transforms the data using a linear operation. The resulting values are then passed through a ReLU (Rectified Linear Unit) activation

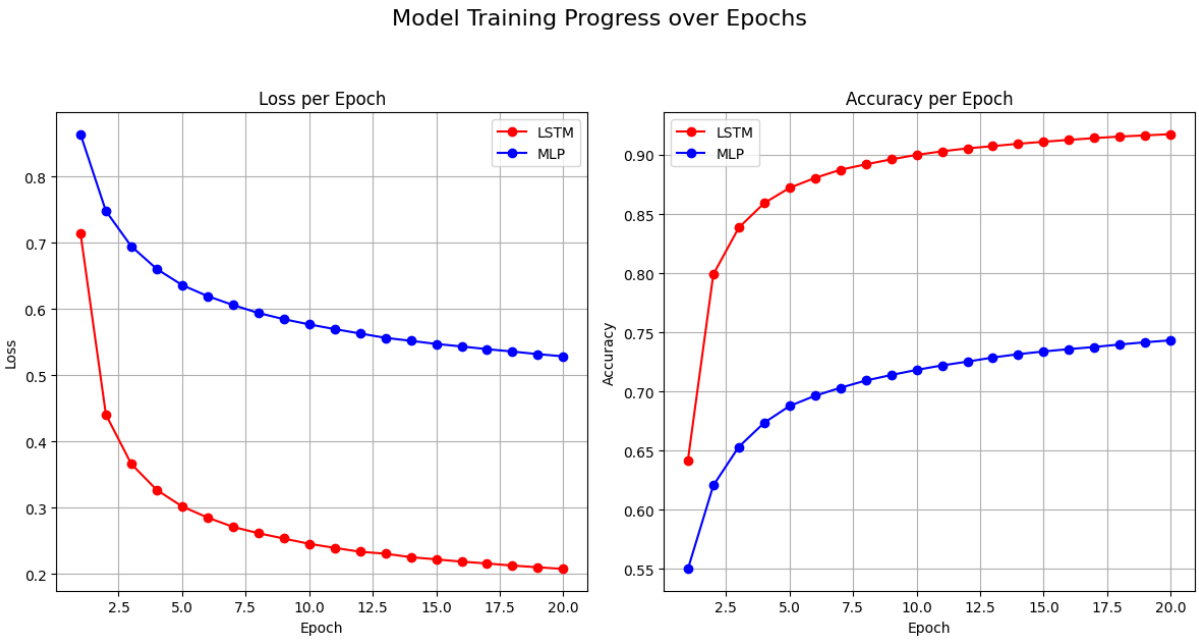

Figure 2: Loss and Accuracy per Epoch for MLP and LSTM Models

function, which introduces non-linearity, allowing the network to learn more complex patterns. The process repeats for the second hidden layer. Finally, the output layer produces the final output and allows it to make predictions based on the input data. Additionally, hyperparameter tuning was conducted on this model as well with different epochs being tested along with different optimizers such as Stochastic Gradient Descent (SGD) and Adam optimizer with Adam and 20 epochs providing the best results of an accuracy of 75.34% as shown in Table 4.

Looking deeper into the loss and accuracy over different epochs as illustrated in Figure 2 shows that for accuracy, the MLP initially increases at a very rapid rate showing that it's learning well from the training data. However, the accuracy curve begins to stagnate around 75%, which is significantly lower than the LSTM's plateau as shown in the graph. The consistent but slight slope towards higher accuracy shows that improvements are being slowly made throughout the training process, but the model may be struggling to capture the complexity of the data. On the other hand, for loss it shows a more gradual decrease in loss over epochs compared to the LSTM. The initial drop isn't as steep which points towards a slower learning rate in the beginning. However, the loss decreases consistently still, without plating as sharply as the LSTM model. This may point to the fact that while the MLP is learning and improving, it may not be as efficient as the LSTM in optimizing the loss function.

The MLP F-1 scores are significantly better than Logistic Regression for all the labels. With scores of 0.72 for Regular, 0.77 for AFib, and 0.74 for Arrhythmia as shown in Figure 5, it shows that the MLP is able to capture the non-linear relationships in a fashion better than Logistic Regression. However, there's still room for improvement as the scores are not very close to 1, but this model meets the initial goal of outperforming logistic regression and presenting scores of greater than 0.7.

For precision and recall, the MLP shows a large improvement over Logistic Regression for the regular label, with a precision of 0.81 and a recall of 0.65 as shown in Table 3. This indicates a high likelihood that the regular heartbeat predictions are correct, and the model captures a majority of the actual regular heartbeat instances. For AFib, the MLP maintains a good balance between precision and recall, with values of 0.71 and 0.77, respectively. This suggests that the model is quite accurate and also fairly comprehensive in identifying AFib cases. The model's performance is also consistent for the arrhythmia label, with a precision of 0.74 and recall of 0.80. This goes to show that the MLP is reliable for arrhythmia detection and identifies a significant portion of actual arrhythmia instances. These values are all very close to each other and consistent, showing

the MLP performs well and is a good option in a healthcare setting - doing a good job in correctly making predictions.

6.4 LSTM

After the MLP, an LSTM was developed that was aimed specifically for our ECG heartbeat data that's presented in a time-series fashion. The LSTM featured multiple layers with the first one processing a individual element of the input sequence at a time and maintaining a memory that captures information about the elements it has processed which allows for the LSTM to remember past data points, which is very important for tasks such as this where the context of the data in terms of time is vital. Following this layer we have a dropout layer, which helps in preventing overfitting by randomly setting a fraction of the input units to 0 at each update during training time. After this, we employ a Dense layer with ReLU activation. This layer aims to understand the features collected by the LSTM layer and begin to process and classify the input sequence into various categories. The final Dense layer has a softmax activation function and outputs the probabilities of the input belonging to each class. We then compiled the model with the categorical crossentropy loss function, which is suitable for multi-class classification tasks, and the Adam optimizer, which is an efficient stochastic optimization method that adjusts the learning rate during training.

All of this leads to the highest accuracy captured in this experiment of 93.04% as shown in Table 4 which backs up the fact of how the LSTM is best for time series data especially because of its memory properties. For F-1 scores the LSTM outperforms both Logistic Regression and MLP, with F-1 scores of 0.89 for Regular, 0.87 for AFib, and 0.90 for Arrhythmia as shown in Table 5. These scores back up the idea that the LSTM's ability to remember long-term dependencies and sequence information in the data make it well-suited for this task. It's particularly strong in detecting Arrhythmia, which may be due to the fact that the LSTM can effectively detect the irregularities between R-R waves causing it to perform so well.

Looking at the loss and accuracy over different epochs, the LSTM shows a very good performance in terms of loss reduction and accuracy growth as shown in Figure 2. Starting with a relatively high loss, it quickly descends within the first few epochs, suggesting that the model quickly learns from the training data. As the epochs continue to grow, the loss curve begins to stagnate, hinting that the model has achieved its optimal loss value which is typical of effective learning, where initial gains are substantial, followed by smaller and smaller improvements as the model approaches its optimum. The accuracy works oppositely, as it rises sharply in the early epochs which shows the model's ability to generalize well from the training data. The accuracy then plateaus near 90% as the epochs continue to expand, indicating that the LSTM has maximized its performance on the training dataset and suggesting that the LSTM is a robust model for this task, perhaps benefiting from its ability to capture temporal dependencies in the data.

The LSTM shows excellent performance for the regular label, with high precision and recall scores of 0.90 and 0.86, respectively as shown in Table 3. This shows that not only are most predictions of regular heartbeats correct, but also that the model is able to recognize the vast majority of true regular cases. For AFib, the model shows even better recall (0.93) than precision (0.82), implying that it's highly capable of identifying most of the actual AFib cases, with some false positives however. The model achieves its best precision on the arrhythmia label at 0.92, with a recall of 0.88. This amazing performance indicates that the LSTM is highly accurate in its arrhythmia predictions and very rarely makes mistakes for the arrhythmia instances. This shows that the model is the best for the clinical setting as it is making the best predictions here with the least number of false positives which can cause issues in our healthcare setting.

6.5 Classical Transformer

In the context of implementing a transformer architecture for ECG data, an important aspect was preprocessing the data to capture the temporal dynamics of 6,428 patients, each characterized by 700 time points. Unlike the typical input format in natural language processing (NLP) transformers, which is conventionally represented as (sequence length, vocabulary size), the input shape for the transformer model structure is designed to be (number of time points, number of features/ECG leads). Instead of having about 4.5 million data points as the other models, the data is now compressed to 6,428 data points. This reduction in data volume made it necessary for a 90%-10% train/test split.

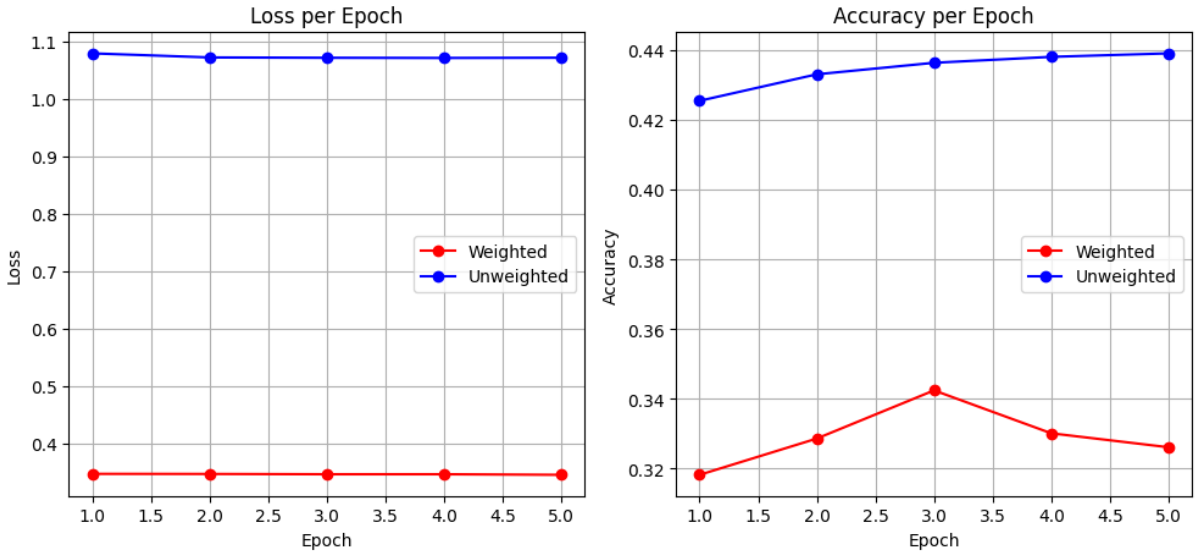

Figure 3: Loss and Accuracy per Epoch for Transformer Models

Once the standard sine/cosine positional encoding and linear embedding are applied to the ECG data, the demographic data is also linearly embedded with equal dimensions so it can be concatenated with the ECG data. This combined input is then channeled through either a self-attention mechanism, in the case of a single layer, or a multi-attention mechanism, when dealing with more than one layer. The subsequent layers are the normalization and dropout layers, introduced to enhance the model's robustness and prevent overfitting. The normalization layers serve to stabilize the training process, while dropout layers contribute to regularization by randomly excluding certain connections during training.

At the end of the architecture, a standard MLP is introduced, contributing to additional depth, nonlinearity, and complexity to the model. This final layer allows the model to discern nuanced waveforms and hopefully improve it's predictive capabilities. The MLP generates individual output values corresponding to each of the three classes. These raw output values are then passed through the softmax function that outputs the probabilities that the sample belongs to for each of these classes.

Similar to the LSTM and MLP architectures, the transformer model also utilized the Adam optimizer and the cross-entropy loss. After numerous iterative experiments, an embedding dimension of 256, a batch size of 512, a dropout rate of 25%, a training duration of five epochs, a single transformer head, and one encoder layer were chosen to be the model's hyperparameters so that it can achieve a balance between computational efficiency and model efficacy.

As illustrated in Figure 3, both the regular and balanced transformer models exhibit minimal divergence in training loss and accuracy across the five epochs. The regular transformer reaches an accuracy of 46.47% and the balanced transformer of 33.40%. This trend implies a limited learning capacity of the model during the training process. Specifically, the regular transformer failed to discern nuances amongst the three ECG waveforms, consistently predicting arrhythmia for all test samples. A closer examination of the label distribution in Figure 1 confirms that the model predominantly opted for predicting the most prevalent label, i.e., arrhythmia.

In response to these limitations, we experimented with weighted loss in hopes of balancing the label distribution and for the model to also learn the other labels. Based on the bar plots shown in 4, the model was able choose other labels, but it often chose the label it became the most accustomed to during its training. In the case of the bar plot, for this exact iteration, it is the regular label. This can be further corroborated by the results in Table 3, Table 4, and Table 5. In interpreting the precision, and recall results, it's apparent that the model performs well on the regular class, as reflected by a precision of 0.94 and a recall of 0.35. However, for the AFib and arrhythmia classes, the precision values are notably lower at 0.02 and 0.01, respectively, indicating a higher rate of false positives. The recall values for

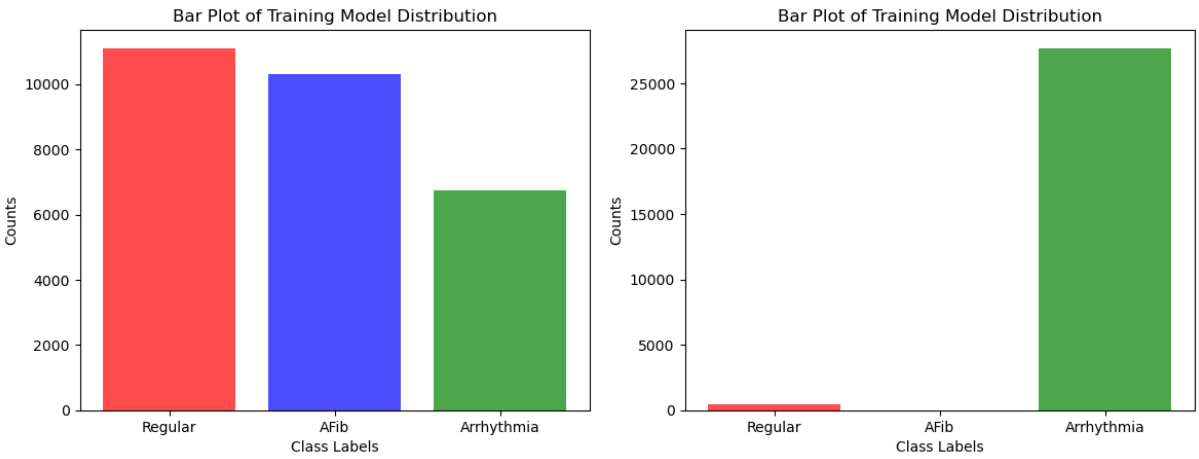

Figure 4: Predicted labels for the classified transformer, left model is the weighted transformer and the right one is the unweighted transformer

these classes are also relatively low. Again, in the F-1 table, the model is correctly identifying half of the regular cases, but only has a F-1 score of 0.04 for AFib, and 0.03 for arrhythmia. From this, it is evident that the model faces difficulty in capturing the subtle distinctions and intricacies inherent in the different types of ECG data labels. The results are highly variable with each run, and there is no stability within the learning of the weighted loss transformer.

From these two models, it is clear that more data is required to achieve a competent transformer model to classify between AFib, arrhythmia, and the traditional ECG waveforms.

7 Discussion and Conclusion

Overall, we experienced the best performance with the LSTM, as we predicted. However, we had some limitations and setbacks with the transformer performance. This is mostly due to data limitations and inadequate compute resources to train the model. Given more time, resources, and data, a future direction can be developing a pre-trained ECG transformer, that can be fine-tuned on tasks such as ours, and can be advantageous for facilities that do not have access to as much data.

The importance of this work is paramount as AFib is generally associated with a significantly increased risk of stroke, heart failure, and other heart-related complications. Early detection of AFib can help with timely intervention, which is vital in mitigating these risks. Furthermore, early detection and management of AFib can significantly improve the quality of life by reducing symptoms like heart palpitations, fatigue, and shortness of breath, which often accompany this condition. In summary, early detection of AFib is a critical step in preventing its progression and reducing the burden of associated health complications, ultimately leading to improved patient outcomes and reduced healthcare costs. More robust models can assist medical professionals in speedier but accurate diagnoses - helping to diagnose AFib earlier in patients and leading to the benefits mentioned.

8 Acknowledgements

Special thanks are extended to Dr. Aleesha Siddiqui, whose insightful expertise in this field helped drive the idea for this project.

References

2023. Atrial fibrillation information. Centers for Disease Control and Prevention. Accessed on November 15, 2023.

2023. Importance of early diagnosis. Get Smart about AFib. Accessed on November 15, 2023.

R. Bousseljot, D. Kreiseler, and A. Schnabel. 2020. Ptb-xl, a large publicly available electrocardiography dataset. *Scientific Data*, 7:1–11.

TR Wei, S Lu, and Y Yan. 2022. Automated atrial fibrillation detection with ecg. *Bioengineering (Basel)*, 9(10):523.

Min-Uk Yang, Dae-In Lee, and Seung Park. 2022. Automated diagnosis of atrial fibrillation using ecg component-aware transformer. *Computers in Biology and Medicine*, 150:106115.