A Deep Learning Network for Detection of Cardiac Arrythmia with Synthetic Electrocardiography

Luke Lorenz & Caroline Cutter

CSCI 701 – Fall 2022

Department of Computer Science

Background

Cardiovascular disease is a serious medical concern due to its high prevalence and mortality rate, accounting for nearly 25% of deaths in the U.S. and costs nearly $229 billion yearly from health care services, medicine, and lost productivity. It is currently the leading cause of death in the United States and continues to increase due to an aging population.

Early detection is vital for treatment and recovery of cardiovascular disease, however, one of the main problems doctors currently face is that diagnosis is done by the naked eye, so early signs of arrhythmias are often missed or misdiagnosed.

Recently, sophisticated deep learning algorithms have proven to offer support to clinical experts for premature medical diagnoses. Raw ECG waveform data contains thousands of data points that are not immediately accessible to physicians. Due to the vast amount of data, deep learning algorithms on ECG waveforms have flourished.

Due to the high incidence rate of cardiovascular disease and inconsistency of diagnosis by eye, the goal of this project is to explore current deep-learning techniques to create a model that successfully recognizes the probability of an arrhythmia using synthetic ECG data to aid doctors in diagnosing different cardiovascular diseases.

Methods

ECG data was generated through a variation of Pierre Elias's library for machine learning on ECG waveforms. This was a python adaptation of ECGSYN from the physionet database, a web-based resource designed to support current research of clinical data. The parameters of interest were sampling rate, duration, gamma, mu_hr_1, sigma_hr_1, min_noise_1, max_noise_1, t, b, and a (Table 1). Using this system, we generated 60,000 examples. Neither patient consent nor administrative approval was required, as this data was purely synthetic.

Our model is an innovative residual neural network. It was implemented using the Keras framework with a TensorFlow backend. Our input shape was of (2500,12). The longer axis (2500) represented the axis of waveform, and most efforts were directed towards convolutions to extract morphological features, while the shorter axis represented the lead axis and was used to fuse data from all leads.

Our model was composed of 4 residual blocks (Figure 2). Each residual block consisted of two blocks which contained a batch normalization layer to normalize data distribution, a ReLU activation function, and an Add to combine the layers (Figure 2). The first unit in each residual block contained a skip connection which allowed skip connections. After the last residual block, our input was fed to a global average pooling layer, which fed to a densely connected neuron and a subsequent prediction.

Our model was trained on a MacBook Air (M2, 2022), with the M2 chip and 8 GB of memory. We acknowledge this hardware as a major limitation to our research. While training, we used an Adam optimizer with an initial learning rate of 0.001. After the development of our model, we input feature data and ECG raw signal of each patient in the validation data into the developed algorithms. We used binary cross entropy as a loss metric, with binary accuracy and AUC as additional metrics.

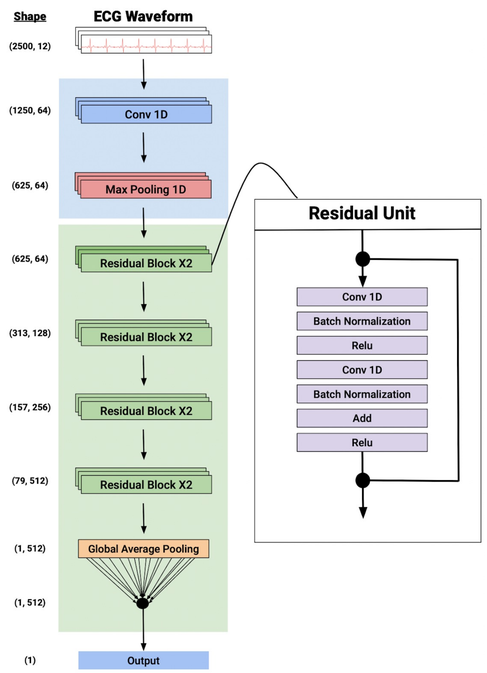

Figure 1. Model structure developed to detect cardiac arrhythmia. Inputs are a synthetic ECG waveform (12 leads & 10 seconds at 250 Hz) of shape (2500,12). Outputs binary prediction of abnormality.

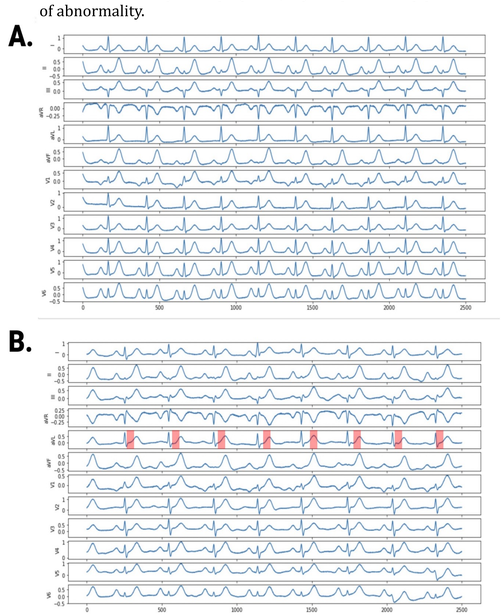

Figure 2. Examples of 12-lead synthetic waveforms. (A) presents a typical read. In (B), red highlights display mild ST segment elevation in the aVL lead.

Parameter	Meaning
sampling_rate	Sampling rate, default 250 Hz
duration	Default 10s
gamma	A (12,5) matrix to modify each lead's five spikes' amplitudes
mu_hr_1	The mean of the heart rate
sigma_hr_1	The variance of heart rate
min_noise_1,max_noise_1	The max value and min value of noise
t	The starting position along the circle of each interval in radius
a	The amplitude of each spike;
b	The width of each spike

Table 1. Parameters for waveform generation.

	Binary Cross Entropy	Binary Accuracy	AUC
Miniscule point abnormalities	0.6760	65.8%	0.739
Accentuated abnormalities	0.1248	97.5%	0.989

Table 2. Accuracy metrics among both datasets.

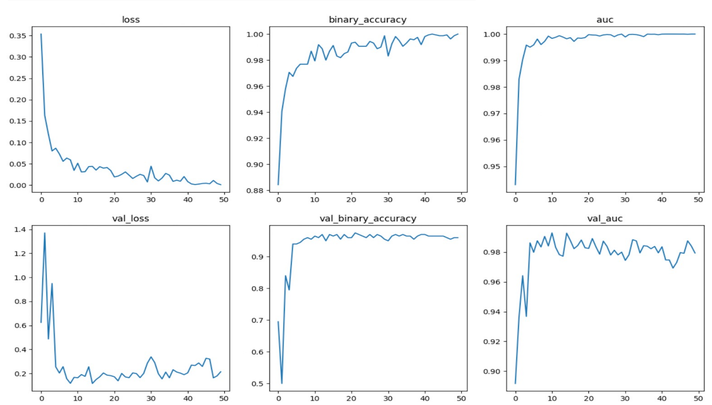

Figure 3. Training metrics for our dataset over 50 epochs. Loss metric is binary cross entropy. Metrics of binary accuracy and AUC were used. Metrics were used on validation sets with 15% of total data.

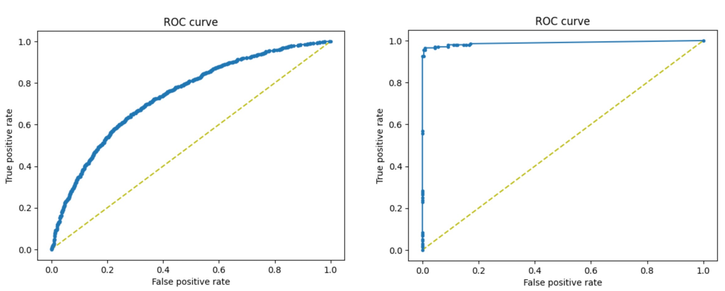

Figure 4. ROC curves for both datasets. The top curve (AUC = 0.738) corresponds to miniscule point abnormalities in leads and the bottom corresponds to accentuated abnormalities (AUC = 0.98)

Results/Conclusions

We trained the model with unique 12-lead ECGs for 25.5 hours. When our model is applied to an ECG, it outputs a binary prediction score, with a value of 1 corresponding to an abnormal waveform and a 0 signifying normal activity. During internal validation, the metrics of our algorithm were improved over 50 epochs of training (Figure 3). While binary cross entropy continued to decrease over the 50 epochs of training, binary accuracy peaked around 25 epochs, which is the number of epochs we decided to train our final model on.

We tested our model with 85% of random examples from our dataset, split 70% testing and 15% validation. The performance of our algorithm was confirmed using a test set of 15% of our data.

For further exploration, we tested our model on accentuated waveforms, or ECG's that had abnormalities noticeable to the trained eye. For this data, our model achieved an accuracy of 97% (Table 2), and an AUC of 0.98 (Figure 4) after 50 epochs. We found this result was promising and detection of more obvious abnormalities was effortless. While our model accuracy was slightly lower for our first dataset, as abnormalities became more obvious our model excelled.

Applications

Classically, barriers to entry in machine learning in medicine have been incredibly high. Patient info is secure and available to healthcare providers, making the nature of machine learning in healthcare/medicine rather secluded behind closed doors and high walls. In this paper, we hoped to illuminate an alternative to these hurdles. As we have seen before, when synthetic data is used in tandem with real world health data, it can be leveraged to provide meaningful premature classification of many sly problems.

Our project can further extend towards more robust classification. Rather than binary classification, outputting a risk score between 0 and 1 of how likely a patient is could assign soft probabilities of likelihoods and establish a pecking order for priority. As medical doctors oversee dizzying numbers of patients, doing this would allow resource priority to go to those with the highest risk.

A second feature to improve our model would be to transition to a categorical classifier of different conditions. Currently, our model only detects abnormalities, but being able to compartmentalize and predict certain conditions is advantageous.

Acknowledgements

This work was made possible by computational resources available from Pierre Elias, M.D. and his repository. See the attached paper for additional sources and references

A second huge thanks towards Professor Basu his guidance and expertise throughout the semester.

The code is available open source at:
https://github.com/llorenz29/ECGNN