HeartNN:

A High-Accuracy Neural Network and Smartphone App for Cardiac Auscultation



Introduction

- Heart disease is the leading cause of death globally, causing nearly 18 million deaths each year.
- Lack of regular heart screening due to limited healthcare availability, unhealthy lifestyles, or other factors can lead to an increased risk of cardiovascular diseases (CVDs). Early diagnosis of CVDs is crucial in reducing the risk of heart failure.
- Doctors perform a process called auscultation to diagnose potential heart problems. This involves using a stethoscope to diagnose heart sounds as normal or abnormal.
- However, this method can only by performed by a trained professional in a clinical setting, thus making it infeasible for under-resourced communities where access to healthcare may be limited. This is where machine learning comes in.
- Using the HeartNN model, people will be able to perform auscultation from the comfort of their homes through a smartphone app. By learning from patterns and trends in heart sound data, the model is able to accurately diagnose the user's heart sounds. This can aid in early detection and prevention of heart problems.

Background

- Heart sounds are caused by the opening and closing of certain valves in the heart and can provide valuable information about the health of the heart.
- Normal heart sounds include the S1 and S2 sounds (often called "lub" and "dub", respectively). These sounds are typically regular and evenly spaced.
- Abnormal heart sounds, which are typically murmurs, indicate a potential problem with the heart. They can be caused by valve disorders like mitral regurgitation, heart defects like ventricular septal defect, and other conditions, such as hypertension.

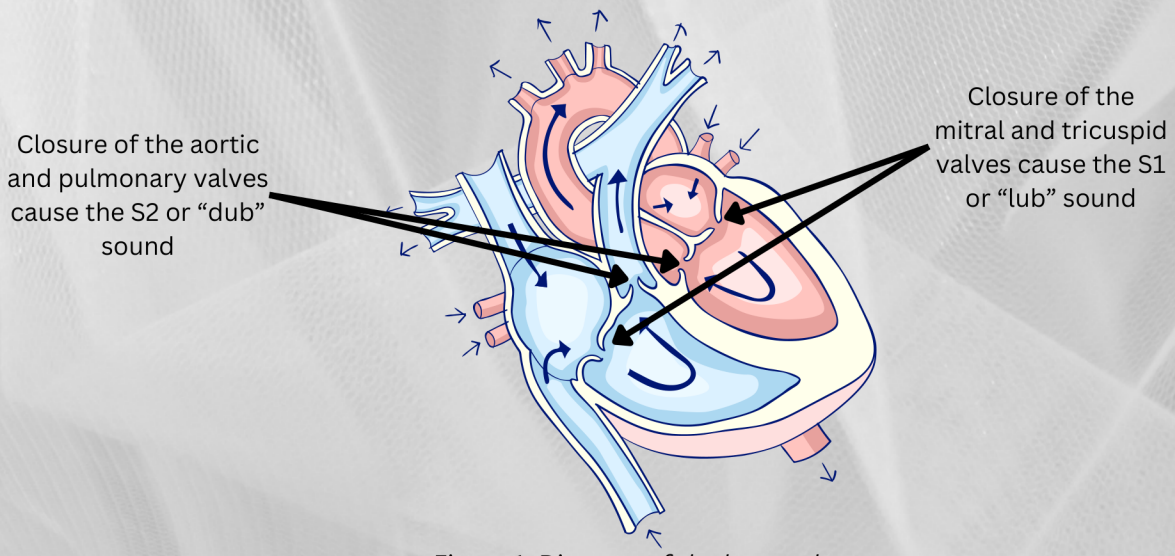


Figure 1. Diagram of the human heart

Engineering Goal

The goals of this project are to:

- Create a robust machine learning model trained on a publicly available heart sounds dataset
- Create an accompanying smartphone app that can accurately classify normal and abnormal heart sounds from user-recorded or uploaded audio.

Training Data

I used the publicly available 2016 PhysioNet/Computing in Cardiology Challenge Dataset, which contains thousands of labeled recordings of heart sounds from both healthy individuals and patients with various heart conditions taken in clinical and nonclinical settings.

Procedure

1. Data preprocessing
    - I first split the audio files into five second segments, then used the Python Librosa library to transform each segment into Mel Frequency Cepstral Coefficients, or MFCCs. MFCCs are a set of features extracted from an audio file. Figure 2, below, shows the Mel Spectrograms and MFCCs for normal and abnormal audio files.

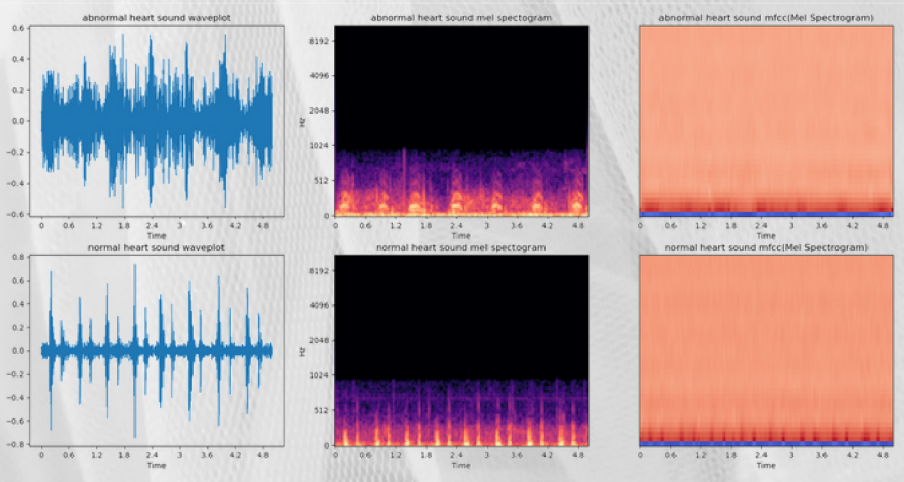


Figure 2. Visualizations of abnormal (top) v. normal (bottom) heart sounds

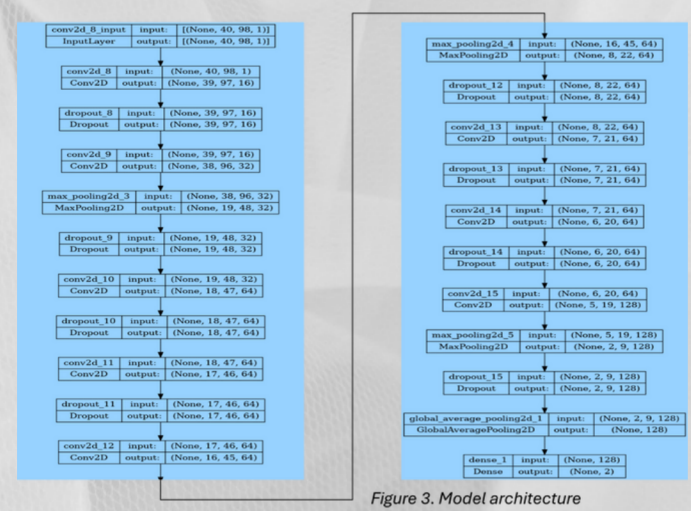


Figure 3. Model architecture

2. Creating the model
    - I used a Convolutional Neural Network (CNN) for this project because it excels at image processing. By treating the MFCC features as an image, I can leverage this for higher accuracy. The full structure of my CNN can be seen in Figure 3 above. It has 22 layers with many different types including convolutional layers, dropout layers, and max pooling layers.

3. Training the model

Before training the model, I split the PhysioNet dataset into 80% training data and 20% testing data.

- Figure 4, on the next page, shows the progression of the model over 500 epochs. The four lines depicted in the graph are:
    - Training loss: This line shows how the model's loss (or error) decreases over time as it is trained on the training set.
    - Training accuracy: This line shows how the model's accuracy increases over time as it is trained on the training set.
    - Validation loss: This line shows how the model's loss on the validation set changes over time and is used to evaluate how well the model is generalizing to new data.
    - Validation accuracy: This line shows how the model's accuracy on the validation set changes over time and is also used to evaluate how well the model is generalizing to new data.

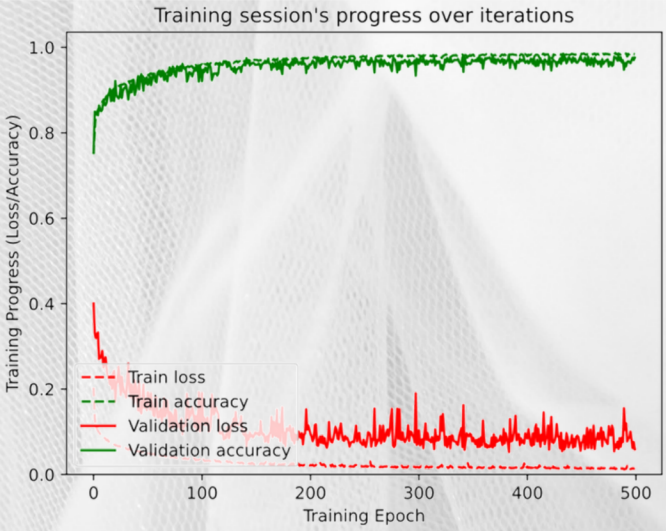


Figure 4. Model training graph

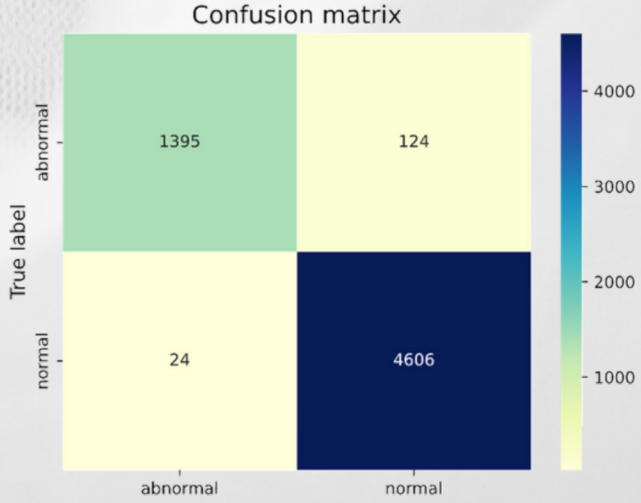


Figure 5. Confusion matrix

Results

- Figure 5, above, shows the confusion matrix for the model. This is a diagram which shows the number of samples the model predicted correctly and incorrectly as well as which class they actually are.
- The model achieved a high accuracy of 97.59%. The challenge score of this model also exceeds the winner of the 2016 PhysioNet CinC challenge, whose dataset I used to train the model.

Smartphone App

After training the model, I saved it and hosted it on a cloud server. I then created a smartphone app in Android Studio which can diagnose a user's sound recording. To use the app, a user can upload their heart sounds onto the app and view its diagnosis. Figure 6 shows this process. This app makes it possible for anyone with a smartphone to perform a preliminary heart screening from the comfort of their own homes. This is particularly important in places where access to healthcare is limited, making heart screening equally accessible for all.

Conclusion

Overall, I built a machine learning model called HeartNN that I trained using the PhysioNet dataset. This model was over 97% accurate in diagnosing heart sounds as normal or abnormal. I then deployed this model on the cloud and developed a smartphone app that used this model to diagnose heart sounds. My project has the potential to improve access to healthcare and reduce the burden on healthcare systems, especially in under-served communities.

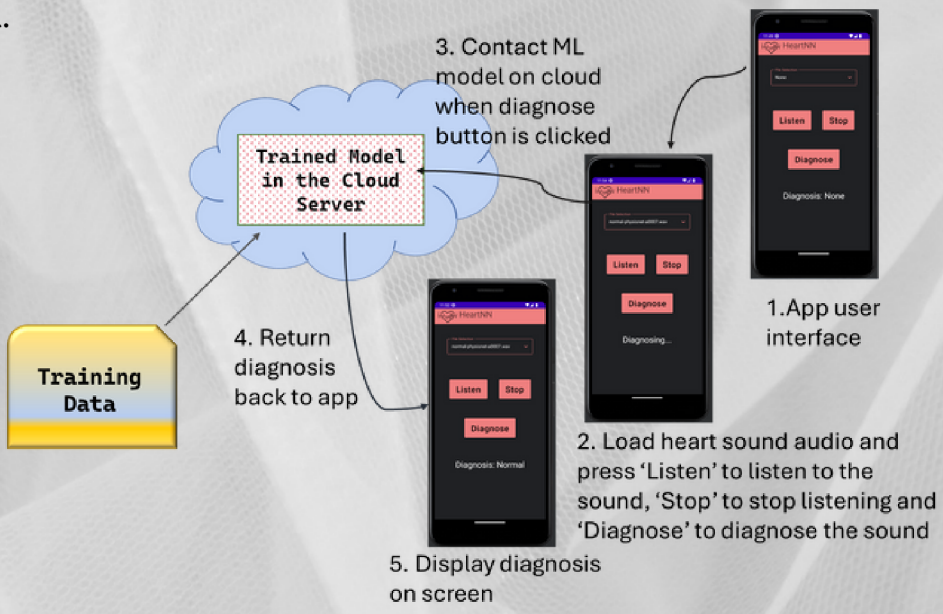


Figure 6 : Integration of ML model on the cloud server with smartphone app

Future Work

As future work, I plan to add more features to the app such as including diet and exercise plans, regular reminders for heart screening, and the ability to share results directly from the app with healthcare professionals.