HeartSense: Detecting cardiac abnormalities using a simple stethoscope recorder and analysis using a deep learning model

Arnav Jhingran

Bellarmine College Preparatory, San Jose, CA

Project Goal

The goal of my project is to design and build a system using off-the-shelf hardware and software to record the cardiac sounds heard from a stethoscope (auscultation) and use a deep learning model to identify cardiac abnormalities.

A simple stethoscope is a physician's most used device for auscultation to recognize heart and lung abnormalities. The cardiac sounds are heard at 4 locations on the chest: i) base right (aortic area), iii) base left (pulmonic area), iv) lower left sternal border (tri cuspid area), v) apex (mitral area). With my project HeartSense:

i) The stethoscope sounds can be recorded on a smart phone using a simple and cheap enhancement to create a combination stethoscope.

ii) A trained deep learning model running on a laptop can quickly classify the sounds and label them to identify any abnormalities. The patient is alerted to seek professional medical help in a timely manner.

iii) The recorded data can be used by a trained medical professional later to validate the findings of the deep learning model and provide feedback.

Typically, a trained cardiologist can recognize the heart sound abnormalities to identify murmurs (typically caused by valve disorders) and other cardiac diseases. However, most patients consult with a primary care physician who may not be trained for cardiac issues, especially in remote communities and developing countries access to specialists and medical professionals maybe limited.

My goal with HeartSense is to build a system such that medical professionals or others with limited specialist training can use a simple stethoscope and a smart phone to identify impending cardiac issues and provide cheap and early diagnosis. The patient can seek medical help quicker for serious cardiac diseases and a trained specialist can later validate the stethoscope recording and identify false positives and negatives. This increases access to early diagnosis in remote locations and countries with limited access to trained professionals. It also help diagnose cardiac problems at home or for screening athletes in schools and colleges without a cardiologist present on site.

Design Overview

The project design consists of two parts: i) the hardware prototype to collect and store cardiac sound data using a stethoscope attachment and ii) the software to collect the raw data from the device, data cleaning, data transformation, data analysis and classification using open source machine learning toolkit (TensorFlow and Keras) and a curated data set from Physionet[3] and PASCAL along with testing with field data.

Background: Understanding Heart Sounds

During the cardiac cycle, the heart's electrical activity causes atrial and ventricular contractions of the 4 heart chambers. This in turn forces blood between the chambers of the heart and around the body. The opening and closing of the heart valves corresponds to the acceleration and deceleration of the blood flow, giving rise to vibrations heard as heart sounds and murmurs. These vibrations are audible at the chest wall, and listening for specific heart sounds can give an indication of the health of the heart. The phonocardiogram is the graphical representation of a heart sound recording captured by a digital stethoscope. Figure 1 illustrates a short section of a Phonocardiogram (PCG) recording.

Auscultation is done at 4 chest locations which are named according to the positions where the valves can be best heard:

- Aortic area - centered at the second right intercostal space.
- Pulmonic area - in the second intercostal space along the left sternal border.
- Tricuspid area - in the fourth intercostal space along the left sternal edge.
- Mitral area - at the cardiac apex, in the fifth intercostal space on the midclavicular line.

Fundamental heart sounds (FHSs) usually include the first (S1 or lub) and second (S2 or dub) heart sounds (Figure 1). S1 occurs at the beginning of isovolumetric ventricular contraction, when the mitral and tricuspid valves close due to the rapid increase in pressure within the ventricles. S2 occurs at the beginning of diastole with the closure of the aortic and pulmonic valves. While the FHSs are the most recognizable sounds of the heart cycle, the mechanical activity of the heart may also cause other audible sounds, such as the third heart sound (S3), the fourth heart sound (S4), systolic ejection click (EC), mid-systolic click (MC), diastolic sound or opening snap (OS), as well as heart murmurs caused by the turbulent, high-velocity flow of blood.

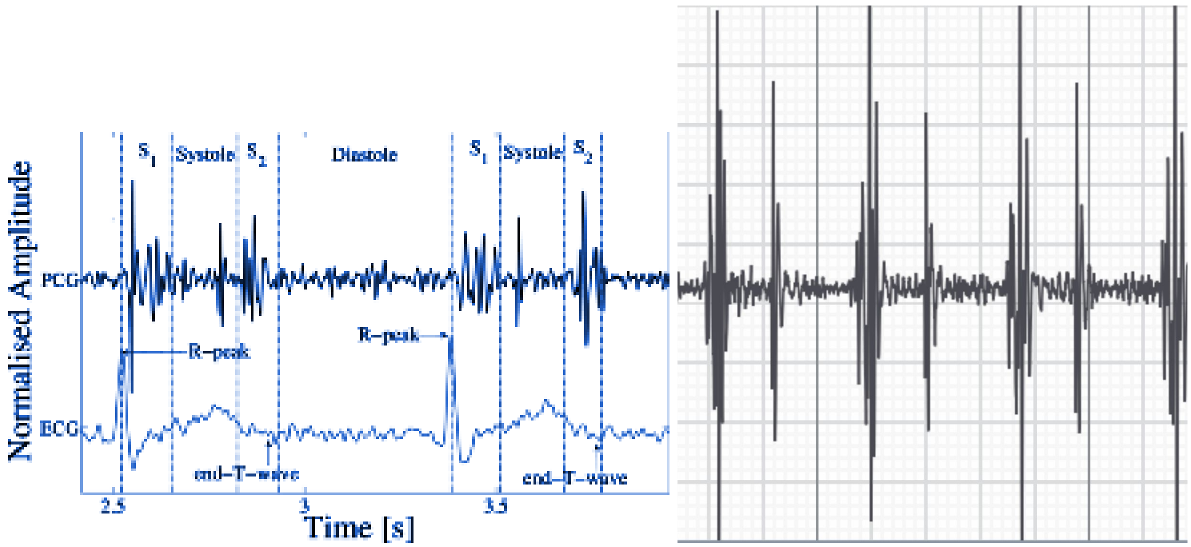


Figure 1: A phonocardiogram (PCG) view of the heart sounds along with the ECG recording below. The sounds are segmented into S1 (lub), systole, S2 (dub) and diastole which are called the 4 fundamental heart sounds (FHS). Image courtesy: Computing in Cardiology challenge, https://physionet.org/challenge/2016. The image on the right is a field PCG recording using my stethoscope attachment prototype showing the S1 and S2 heart sounds.

The segmentation of the FHSs is a first step in the automatic analysis of heart sounds. The accurate localization of the FHSs is a prerequisite for the identification of the systolic or diastolic regions, allowing the subsequent classification of pathological situations in these regions.

Abnormal and Normal Phonocardiograms

Figure 2. Simplified FHS on a phonocardiogram with common abnormalities. A) Normal heart pattern B) Acute Mitral Regurgitation C) Mitral Regurgitation due to Coronary artery disease. Image courtesy: http://www.med.umich.edu/lrc/psb_open/html/repo/primer_heartsound/primer_heartsound.html

The images (Figure 2) above detail the murmurs diagrams based on certain heart conditions heard from a supine position. The S1 and S2 (lub and dub) are always present and are diagrammed into large rectangles for simplicity. The pattern of the murmurs in the Systole is what is used to determine certain Mitral conditions.

These FHS and the abnormalities are detected using a stethoscope that can digitally record the sounds.

Hardware Design Overview

A traditional stethoscope consists of a i) chest piece with diaphragm, ii) bell, iii) tubing and iv) headset. The tubing attaches to the chest piece at the stem and can be removed. My design removes the chest-piece and attaches it to a small tubing containing a microphone that connects to the smart phone audio jack. The original tube can be attached back after the measurement is taken. I used software to filter the noise basically reduce the extraneous sounds in the recording and eliminate sounds that are out of the range of any possible heart sound (typically under 200 Hz).

Figure 3 (a, b): Normal stethoscope modified for digital recording. a) On the left is a simple tube with microphone attachment. b) On the right is the same attachment with a mic splitter to attach the headphones to hear the sound played back from the smartphone.

When using the modified stethoscope (Fig 3a) to record heart sounds the main problem is how to hear the heart sound correctly before starting the recording. A splitter and earphone jack was used to playback the mic input back through the headphone using audible (an open source sound mixing software). The problem in this approach was the doctor taking the recording was hearing the sound played back from the mic input directly and not through the stethoscope headset. In early feedback from a cardiologist this approach was not considered optimal for medical professionals who are trained for stethoscope usage. Moreover the sound digitally amplified or directly played hears different from what is normally heard through a stethoscope.

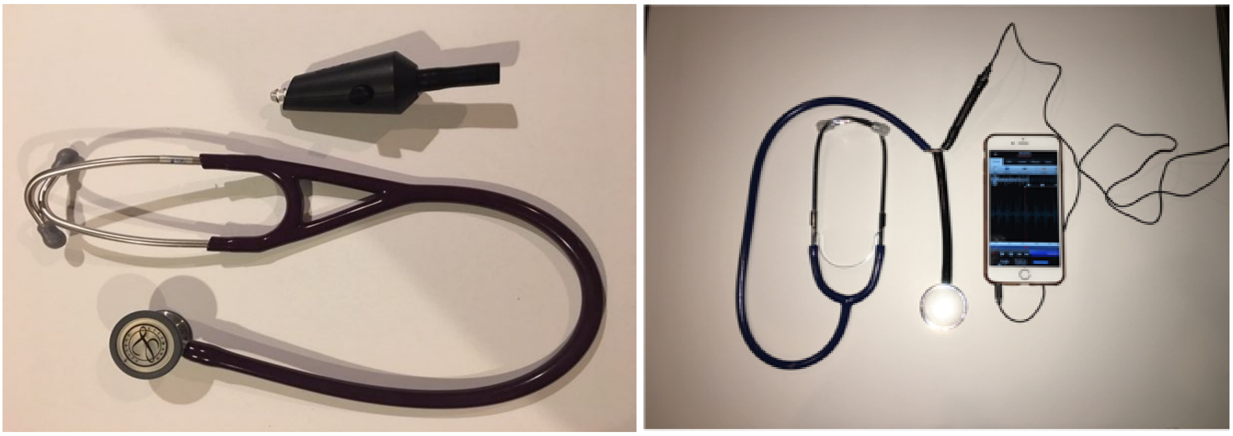


Figure 4 (a, b): Combination stethoscope for digital recording and direct hearing a) On the left is an Eko device, a commercial digital stethoscope attachment b) On the right is the prototype of a combination stethoscope using a tube splitter attachment that can be heard directly via the binaural headset and recorded simultaneously via a microphone connected to the audio input of a smartphone.

While exploring other options for digital recording while simultaneously hearing like a normal stethoscope, the design of a teaching stethoscope looked promising. Here the main shaft had an attached splitter for two stethoscopes to be attached. In my prototype (Figure 4b) I used a similar idea and attached a metal tube splitter to the main shaft to enable simultaneous recording and hearing the sound through a binaural headset like a normal stethoscope. In feedback from a cardiologist this approach seemed to work in the field and was much cheaper than the commercial attachment.

Software Design

After the phonocardiogram's raw data file is transferred from the smart phone it is processed in 5 steps: i) Training data collection ii) Field data Collection for testing data iii) Data Cleansing and Standardization iv) Data Transformation and v) Data Analysis using deep learning model trained and validated on a curated dataset.

Training Data Set

The training data used is based on a publicly available curated dataset of 3,126 cardiac sounds as part of the Physionet computing in cardiology challenge 2016 (https://physionet.org/challenge/2016/) and 86 cardiac sounds (after eliminating artifacts) from the PASCAL challenge (http://www.peterjbentley.com/heartchallenge/).

The heart sound recordings were collected from 4 different locations on the body, the aortic area, pulmonic area, tricuspid area and mitral area, but could be one of nine different locations. In both training and test sets, heart sound recordings were divided into two types: normal and abnormal heart sound recordings. The normal recordings were from healthy subjects and the abnormal ones were from patients with a confirmed cardiac diagnosis. The patients suffer from a variety of illnesses but typically they are heart valve defects and coronary artery disease patients. Heart valve defects include mitral valve prolapse, mitral regurgitation, aortic stenosis and valvular surgery. All the recordings from the patients were generally labeled as *abnormal.* The data does not provide more specific classification for these abnormal recordings. Both healthy subjects and pathological patients include both children and adults of either gender. Each subject/patient contributes up to six heart sound recordings. The recordings last from several seconds to up to more than one hundred seconds. All recordings were resampled to 2,000 Hz and available as .wav format file. Each recording contains only one PCG lead.

In the PASCAL challenge, data was classified as normal, murmurs or something called EHS (extra heart sound). Medical opinions vary on the dangers of extra heart sounds but with a large amount of them occurring from benign causes, EHS can be labeled as a normal heart sound. The Physionet data was only labeled as murmur or normal meaning that EHS was labeled as normal in this data set. In order to be able to use both data sets as training for the model, all EHS labels were relabeled as normal heart sounds. A fraction of the recordings had too much sound and were marked as junk and not used in the training. The union of these two data libraries gave a set of around 3000 heart sound recordings, out of which around 2200 were normal, and 800 were abnormal.

Field Data Collection

I used my combination stethoscope and the commercial Eko bluetooth-enabled stethoscope to collect the heart sounds from 4 chest locations (Figure 4 a) using the bell and diaphragm (Figure 4 b) from the volunteer students in my school, work colleagues of my parents, community center volunteers and some of the cardiology patient’s at Southbay Cardiovascular under the guidance of Dr. Chirala. The data was collected on my phone as a .wav file using Eko’s recorder (for the Eko stethoscope) and the wavepad app (for my combination stethoscope) built using the audible open source audio libraries used for sound recording and mixing. The volunteers were asked to be seated and breathe normally. The readings at Southbay cardiovascular were from 5 known murmur patients. These readings were taken when the subject was lying down and lying on the left side. Other readings from the 25 non-patient subjects (both adult and teenagers) were all seated or standing and not lying down. For some subjects not all chest locations were used for the recordings. 2 readings were taken for most volunteers. Each recording was around 15 to 30 secs.

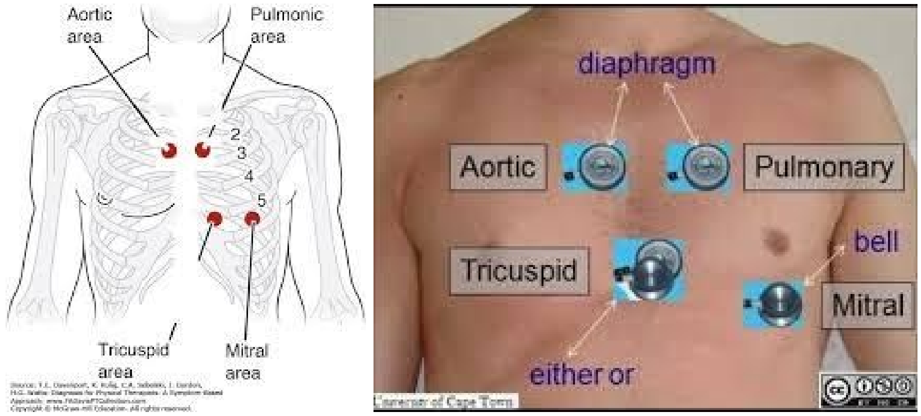


Figure 4 (a and b): Chest location for the heart sound recording with a stethoscope bell or diaphragm or either. Image Courtesy: www.FADavisPTCollection.com, McGraw-Hill Medical.

Data Cleansing and Amplitude Normalization

For each data collection set, for each recording we remove the first and last second to remove any sounds recorded while starting and stopping the recording. A noise filter was applied to the data set to remove sounds outside the range of heart sounds. For each data set we then normalize each data column to eliminate the units and make the data stay within bounds such that the mean is 0 and the standard deviation is 1. For normalization, we compute the mean for each data column and the standard deviation. Then from each value we subtract the mean and divide by the standard deviation, $x_i = (x_i - \mu) / \sigma$

By doing this the different units for the data in the different recordings will not affect the results.

Feature Extraction

The most obvious way to have a machine detect abnormalities is for the model to replicate the process of how a doctor checks for heart abnormalities. A doctor or cardiologist would identify the lubs (S1), and dubs (S2), and then analyze whether any murmurs or extra heart sounds occur between S1 and S2 and back to S1. A machine learning model would then replicate this segmented analysis: identifying S1 and S2 and segments in between, and then identifying characteristics of these segments to determine whether a murmur is present. With a deep learning model, instead of identifying individual segments the model is trained across a lot of data to detect hidden patterns useful for classification. The focus of my project is to build a deep learning model to directly classify heart sounds as normal or abnormal as shown in Figure 5 below

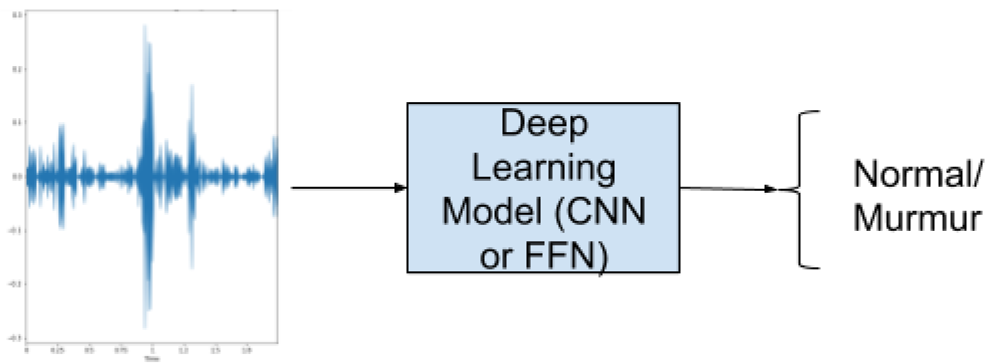


Fig 5: A basic deep learning classification for heart sounds

There are three ways of representing the PCG sound wave:

- A pure time amplitude series {t1:a1, t2:a2, …} which are called the **T features** (Fig 6)**.** Since deep learning models require fixed size features for all inputs, the time series needs to be padded (or repeat) to a fixed size, in this case, 20,000 time intervals. Also it is important to subsample the time features at 400Hz to remove extraneous sounds (heart beats are typically 195 Hz or below, and subsampling at 2X should eliminate most extraneous sounds).

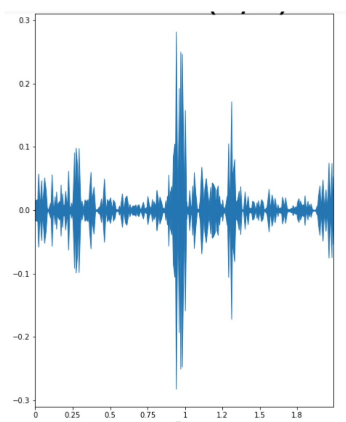


Fig 6: A PCG as a basic time amplitude series

A pure frequency representation can be viewed as a sum of frequencies using a fourier transform (FFT). After the transform the PCG can be represented as {f1:v1, f2:v2, …} which are called the **F features** (Fig 7). There is no need for any subsampling, since the frequency domain can always be restricted, though it turns out that higher frequencies naturally have a very low value (here green and yellow represent high value).

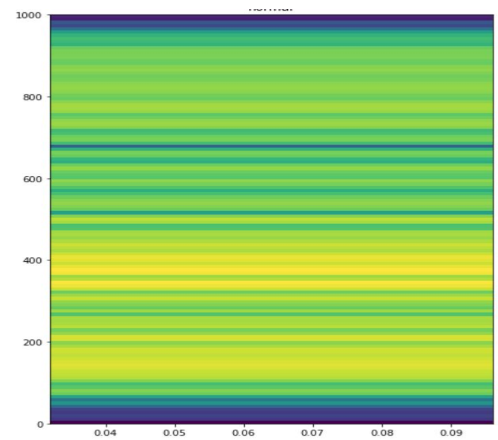


Fig 7: A PCG as a frequency representation using a Fourier transform

- Since segments are still critical, the series can be sliced into smaller time segments, and then determine the frequency features within the segments. This allows for the preservation of the frequency features within the S1 segment, S2 segment etc. without combining all into one. Hence this can be represented by {t1:{f1:v1, f2:v2, ..}, t2:{f1:v3, f2:v4,...}, …}. labeled as **TF features**. As with F features, the frequency domain can be restricted and have no requirement for subsampling. Here one can see that even at 800Hz, the frequency amplitude falls to around -70dB, which represents a 10^-7 amplitude compared to the lowest frequencies.

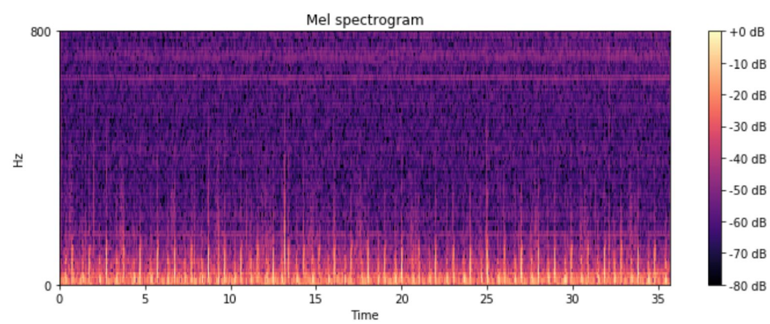


Fig 7: A PCG as a time sliced and frequency representation using a Fourier transform

Deep Learning Models

A neural network is a machine learning technology patterned after the human brain. It consists of an input layer, multiple hidden layers and an output layer. Multiple types of deep learning networks have been researched for various applications from image classification, natural language processing to self driving cars. Some of the models most useful for wave or image classification are generic Feed forward networks (FFN) or a class of them called convolution neural networks (CNN). CNNs are used most commonly for image classification where a filter (a convolution function) is applied to fixed size subset of an image to identify components of an image and then classify the image. Both FFN and CNNs do not have cycles or loops. Other more complex network types like Recurrent neural networks (RNN) and long short term memory (LSTM) have loops and hidden state that is useful to "remember" previous input but do not apply that well to wave and image classification.

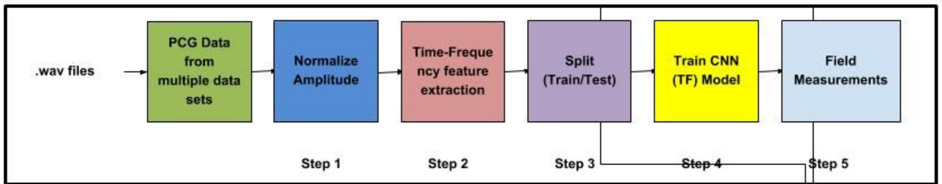
For the PCG classification, based on previous research, I explored the following algorithms for the time and frequency features:

1. A Recurrent Neural Network, preferably an Long Short Term Memory (LSTM) model for **T features.** However, research suggests that a Convolution Network (CNN) might work better. Long sequences do not typically work with LSTM, and at a sample rate of a 1000 Hz, a 30 sec heartbeat sound would have 30,000 time sequences, making LSTM less effective. A CNN for **T features** is preferable.
2. A fully connected feed forward network (FFN) for single dimension representations, such as **F features.**
3. A Convolution network (CNN) for multi-dimension representations, such as **TF features.**

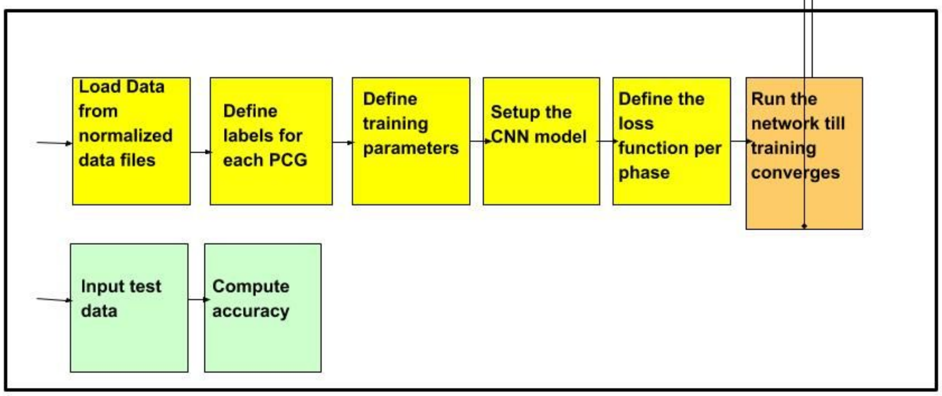
Experimental Setup and Methodology

Data Analysis and Classification with Tensor Flow and Keras library

TensorFlow (https://www.tensorflow.org/) is a software library in Python, developed by the Google Brain Team within Google's Machine Learning Intelligence research organization, for the purposes of conducting machine learning and deep neural network research. Neural networks can be used for machine-learning based classification and TensorFlow provides some predefined neural networks like CNNs that I use as my base to extend upon. With TensorFlow as the backend, I use the Keras library to simplify programming the model. Keras is an open source neural network library also in Python that is commonly used for fast prototyping. The entire flow of the steps used to classify the heart sounds is shown in the following flowcharts:



Flow Chart of PCG classification to normal or abnormal heart sound



Flow Chart for Step 4: “Train CNN Model with Time and Frequency features”

I used the following software packages for various stages of data processing.

1. Use wavepad software on the iphone to download the PCG as .wav file. Also the Eko app was used to get the recordings made through the Eko bluetooth enables stethoscope attachment.

2. `Anaconda 5.0` Python distribution which includes the `Jupyter` notebook and the `Spyder` Python dev environment.

3. In Python included the the libraries: `matplotlib` (to plot the data in the intermediate stages), `numpy` (package for various scientific and statistical functions) and `TensorFlow 1.0.0` (the neural network package) and the Keras library (https://keras.io/).

Experimental Results

The modified accuracy value of > 0.8 will be considered acceptable. The modified accuracy measure is the average of the true positives and the true negatives as detected by the model. The training and validation accuracy of the three algorithms (CNN-TF, CNN-T and FFN-F) are given below:

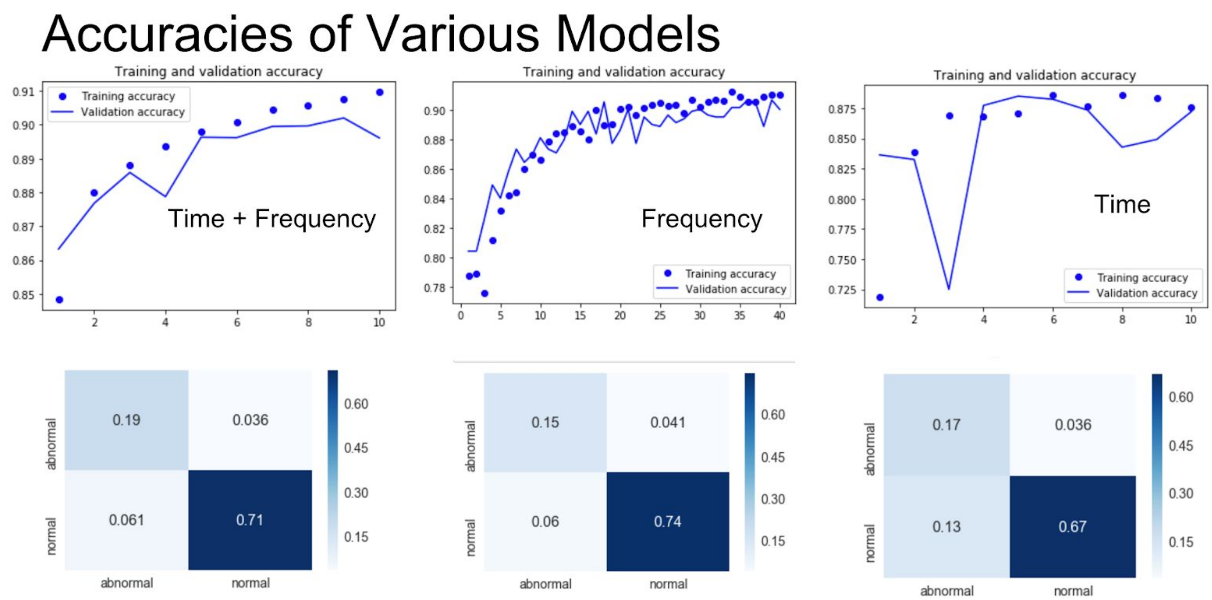


Fig 8 (a,b,c): The training and validation accuracy of a) CNN (Time, frequency), b) FFN (frequency) and c) CNN (time). The confusion matrix is shown below.

The first sets of rows show two plots--how the training of the model happens as it runs, and how at each epoch (one pass through the entire data set), the model performs against the test data set. In general, models train will against the training data set (seen repeatedly), the real test is how the model do against the test data set (validation set). As seen in Figure 8, the validation accuracy for the CNN-TF and FFN-F are near 90%, and the validation accuracy of the CNN-T model is near 87.5%.

The second set of rows shows the confusion matrices of the three models. Columns are when the data is truly abnormal or normal, and the rows are the predictions from the models. A model with good accuracy will have the weights entirely along the leading diagonal (top left to bottom right), but as we can see, some fraction of heartbeat sounds are misclassified.

The accuracy in abnormal sounds (false negatives) is more important--missing abnormal sounds has a higher penalty in real life than classifying a true normal as abnormal. The CNN-TF model has lower false negatives compared to the others.

Field Data Testing Results

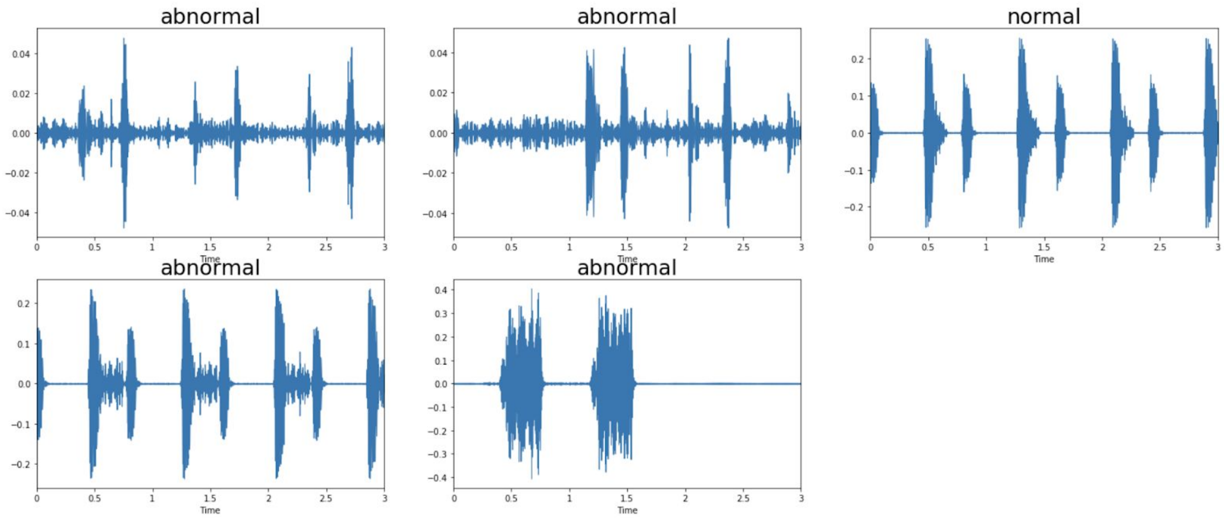


Fig 9: The sample of field data PCGs that are all labeled abnormal by the specialist. In this sample set the CNN-TF misclassified one as normal.

The TF model did reasonably well, but it did misclassify some abnormal PCGs as normal ns vice versa with an accuracy of ~80%.

Out of the recordings from 30 different subjects normal heart sounds that were hand collected, Figure 9 and 10 shows the sample of 5 abnormal and normal PCGs.

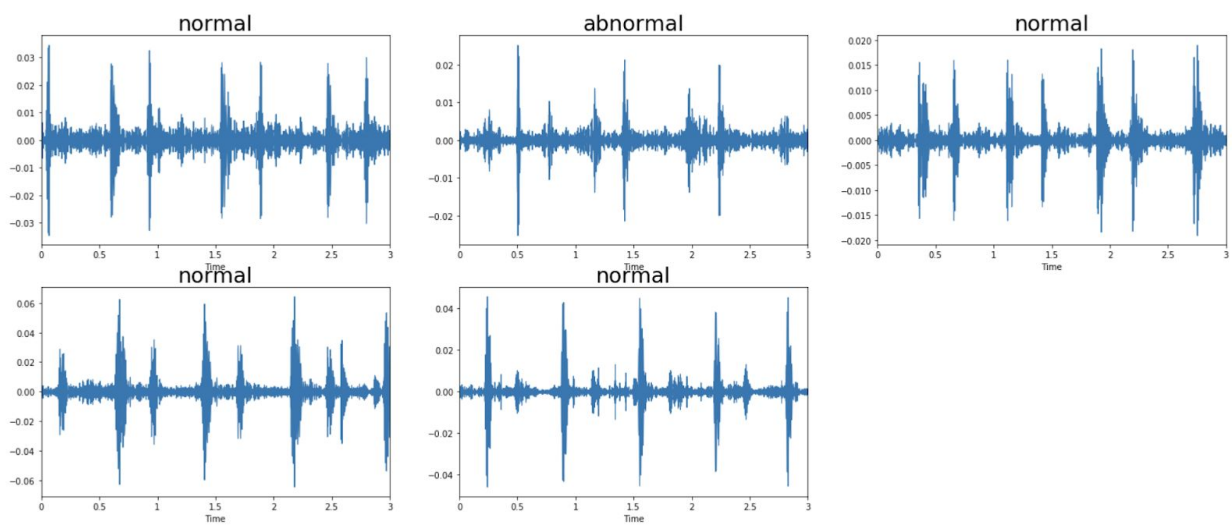


Fig 10: The sample of field data PCGs that are all labeled normal by the specialist. In this sample set the CNN-TF misclassified one as abnormal.

Although the TF model did well compared to the other models, even though it misclassified some of them, it did worse than the accuracy of the validation data sets. Those were all data collected from the same recording methods and labeled together, the field data was collected by different hardware and labeled only for murmurs and not other heart sounds.

Stethoscope Comparison Results using Field Data

Next we compare the machine learning predictions from the Eko device and the cheaper combination stethoscope prototype device. A sample of two subjects, the sound waves for the same patients look different in the two devices in the Figure 11 below.

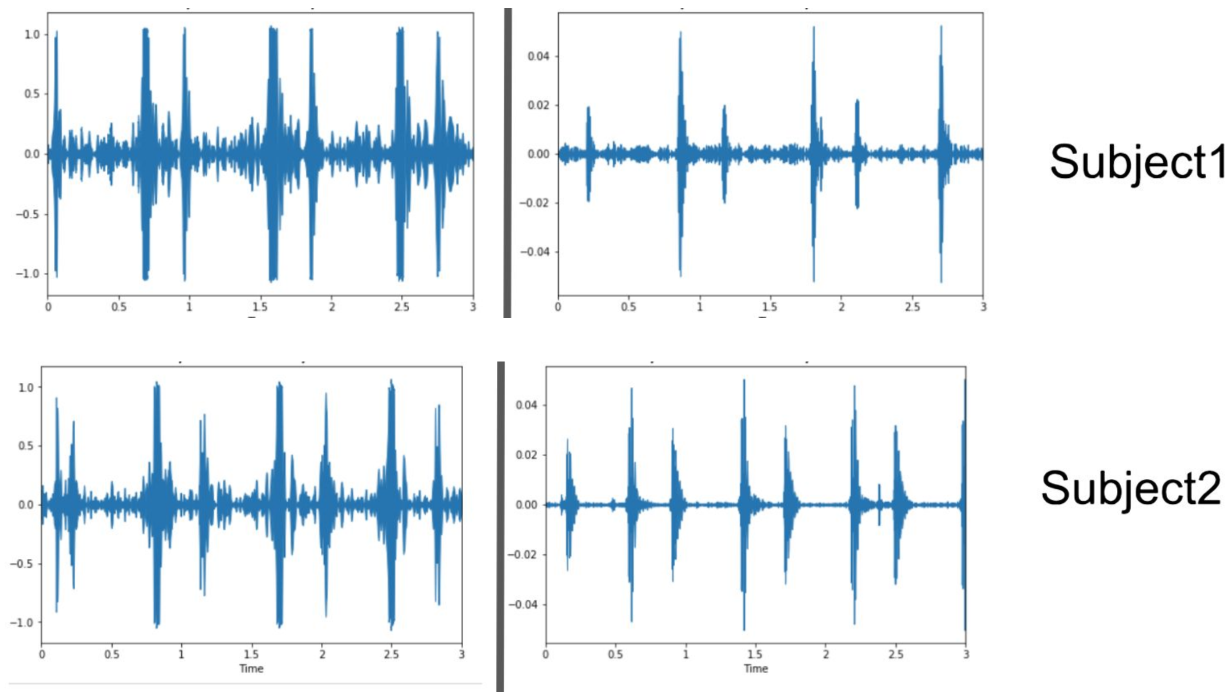


Fig 11: The sample of field data PCGs from the commercial Eko device (left) and combination stethoscope prototype (right) from 2 subjects.

However, surprisingly, the accuracy of the cheaper prototype device was 24/28 (85%), whereas for the Eko device it was only 23/28 (82%). This might be due to the fact that the training data from Physionet and Pascal was recorded using a recording attachment much like my first prototype recorder and the Eko data is amplified digitally and might confuse the trained models. The training data needs to be collected using the Eko device to better understand this behavior.

Challenges and Future Work

One obvious challenge for is collecting enough data from the field with a similar stethoscope so that I can both validate the models, but also validate that my device is working accurately. My preliminary study indicates that it works well but I had recordings from only 30 subjects for the field data and this would need at least a 1000 subjects to be conclusive.

The second challenge is the quality of the algorithms. Doctors determine heart abnormalities by understanding the lub and dub sounds--we did not do any segmentations of the heart sounds to try and determine the lubs and dubs. It is entirely possible (and some researchers suggest that), that time segmentation is a useful first step to classifications.

The third challenge is that not all heart sound abnormalities are created equal. Aortic stenosis is for more serious than an extra heart sound. The current dataset I used only labeled the sounds as abnormal without any specific disorder. There are some new techniques like one-shot learning or k-shot learning that might apply to train against multiple labels where there are not that many training samples for each disorder.

The final challenge is the quality of the stethoscope. My prototype attachment works with any standard stethoscope but the quality of the stethoscope is also important to detect low grade heart murmurs. Digitally enhancing the sound helps but also introduces errors as the models are trained without digital amplification.

In future, I want to explore other body sounds for screening and diagnosis. The stethoscope is used to listen to not just heart sounds, but lung sounds and stomach sounds and my study can be extended into these other areas of the body.

Conclusions

A combination stethoscope is preferred by medical professionals compared to a purely electronic one. Converting the training stethoscope approach to build a combination stethoscope worked well for field data collection and had similar accuracy (~85%) to a commercial one. The choice of neural network was not that obvious initially. A CNN model with time and frequency features worked reasonably well (~89-90%) accuracy. The model was trained with the Physionet and PASCAL data sets which only labeled normal and abnormal heart sounds and did not specify the particular type of murmur. Field data collection and testing showed that some heart sounds labeled abnormal are in fact benign. The dataset collection needs finer labeling and details for more specific diagnosis.

Bibliography

1. Google, TensorFlow. *https://github.com/tensorflow*

2. University of Michigan Medicine, Heart Sound and Murmur Library, http://www.med.umich.edu/lrc/psb_open/html/repo/primer_heartsound/primer_heartsound.html

3. Computing in Cardiology, Physionet Challenge, https://physionet.org/challenge/2016.

4. Siddique Latif , Muhammad Usman , Rajib Rana , and Junaid Qadir, Phonocardiographic Sensing using Deep Learning for Abnormal Heartbeat Detection, https://arxiv.org/pdf/1801.08322.pdf

5. Anand Bhaskar, A simple electronic stethoscope for recording and playback of heart sounds, Department of Physiology, Christian Medical College, Vellore, Tamil Nadu, India, https://www.physiology.org/doi/pdf/10.1152/advan.00073.2012

6. Jonathan Rubin, Rui Abreu, Anurag Ganguli, Saigopal Nelaturi, Ion Matei, Kumar Sricharan, Recognizing Abnormal Heart Sounds Using Deep Learning, https://arxiv.org/abs/1707.04642

7. Sethio, CardioPhonograms App on iOS, https://stethio.com/features/phonocardiogram/

8. Gari Clifford et al, Recent advances in heart sound analysis, Physiological Measurements, 2017, Vol. 38. No. 8.

9. Cardiac Auscultation, Merck Manual, https://www.merckmanuals.com/professional/cardiovascular-disorders/approach-to-the-cardiac-patient/cardiac-auscultation

10. Gari Clifford et al, *Recent advances in heart sound analysis,* Physiological Measurements, 2017, Vol. 38. No. 8.

11. Potes et al., *Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds*, Computing in Cardiology, 2016

12. Logistic Regression-HSMM-Based Heart Sound Segmentation, Springer et al., IEEE Transactions on Biomedical Engineering, 2016

13. Jiang et al., *Music Type Classification By Spectral Contrast Feature*, IEEE International Conf. on Multimedia and Expo, 2002

14. Piczak et al., *Environmental Sound Classification With Convolution Neural Network*, IEEE International Workshop on Machine Learning and Signal Processing, 2015

15. Advanced Deep Learning with Keras, Pocket book, Atienza, 2018.

16. https://github.com/ajhingran/heartsense [my github link, to be uploaded]