Alex Brown, Alex Farfan, Evie Holyoake, Daniel Payan

Team daea

Bi-Weekly Progress Meeting 1

2/15/2022

1. Provide an update on your progress according to your Gantt chart in the proposal; provide over/under of your progress vs plan

We're on week 4 or Phase 2 of our Gantt chart.

For Phase 2B, we found that our interpretation of masks was incorrect, which threw off the timeline of the rest of our progress and led us down other research avenues. Though we have generated masks, they aren't very productive and there is no test we have done that we could use for this project. This means we are behind the progress listed in the proposal under Phase 2B.

For Phase 2A, a filter has been tested, but was found to be unreliable. The time it took to find data has impeded on the ability to produce other filters. MFCC vectors and file production from the Coral Board have not yet been started. This means we are behind the progress listed in the proposal under Phase 2A.

2. Provide an update of teamwork: who is doing which tasks

- Alex Brown & Evie Holyoake
  - ✓ Researched audio data sets
  - ✓ Developed a data preparation script in python
  - ✓ Tested Models
    - Linear Regression
    - K-Neighbors Regression
    - Decision Tree Regression
  - ✓ Neural Network Test
  - ✓ Gaussian Mixture Model Testing
    - Gender Separation
  - ConvTasNet Research
  - K-means "soft" clustering for initial separation
  - Set up meeting with Professor Bell to discuss neural networks
- Alex Farfan & Daniel Payan
  - ✓ Met with Professor Mudumbai and Professor Kruger
    - Discussed removal of background noise instead of reduction
  - ✓ Found audio data sets for machine noises
  - ✓ Plotted frequency and time domain of audio segments
  - Implement two different filter models
    - Notch Filter
    - Match Filter

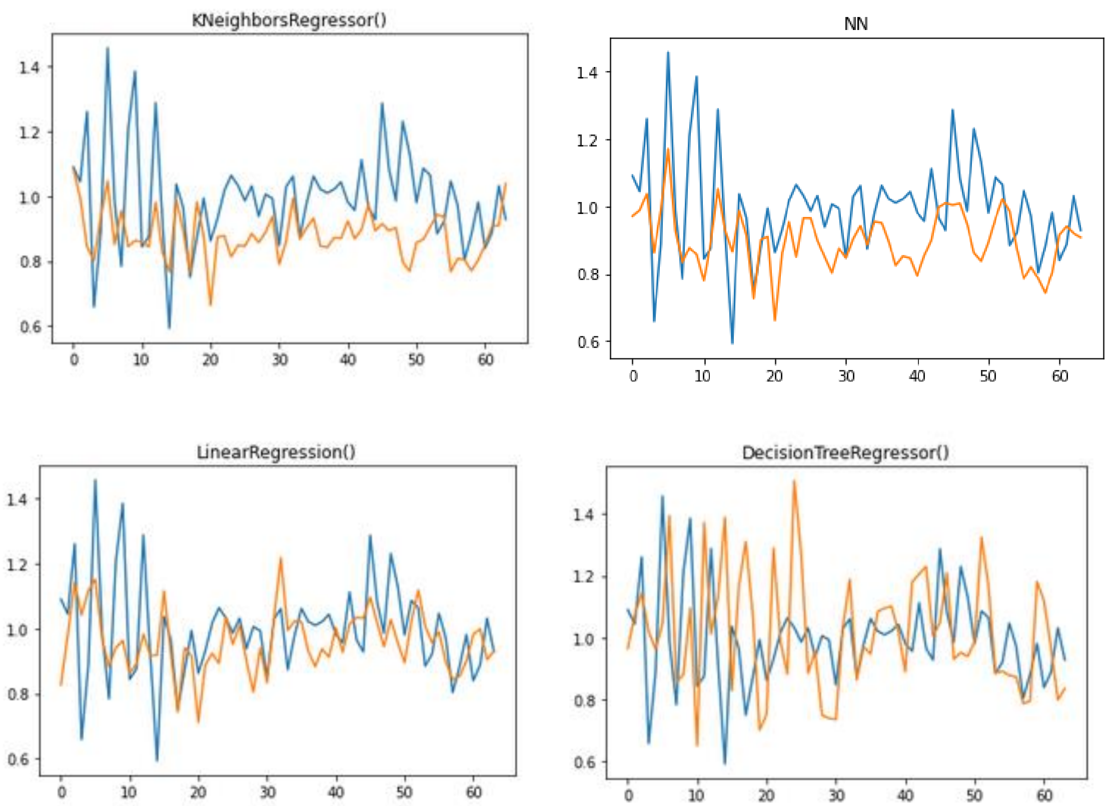


Figure 1: Initial tests to verify input/output dimensions (blue is target, orange is actual)

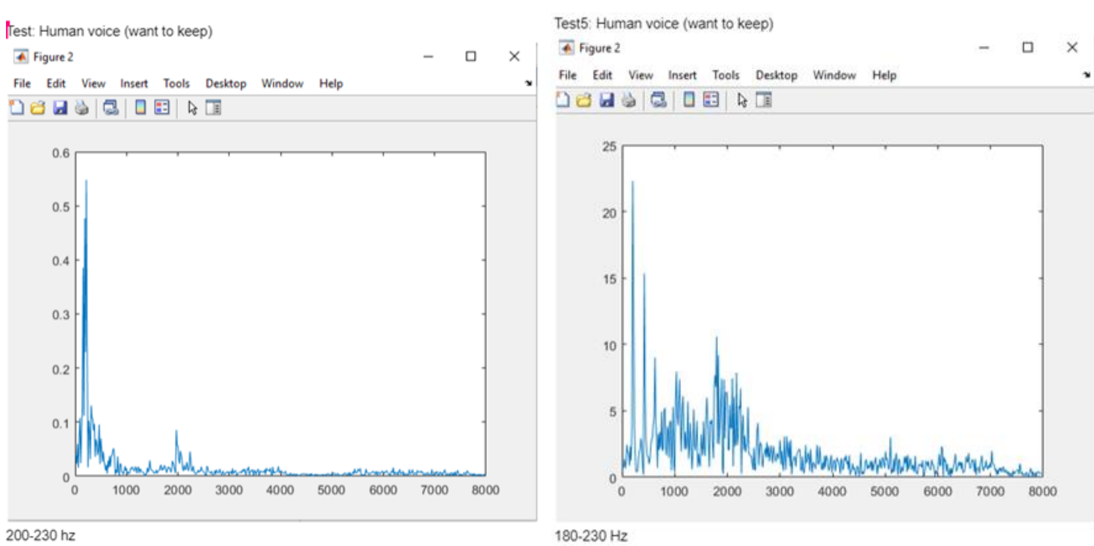


Figure 2: Frequency domain of human voice

3. If any new bottleneck/hurdle is discovered, please describe; also describe your solution to address the hurdle

- ☹ Our sponsor isn't able to provide data
    - This has caused issues with creating guidelines for initial filters as we have had to guess on frequencies that will be encountered
    - This has also caused issues because we had to spend a lot of time sourcing data sets with the sounds that are expected to be filtered out
- ☹ Our sponsor is unable to provide guidance on some aspects of our project
    - This has created a lack of direction (or really too many directions with no focus)
    - Due to the research and development nature of our project, we are finding that our initial proposal isn't feasible
- ☹ We are all inexperienced in this field
    - Though papers can be found accomplishing our task, these developments aren't publicly available and they hold higher qualifications with longer time lines

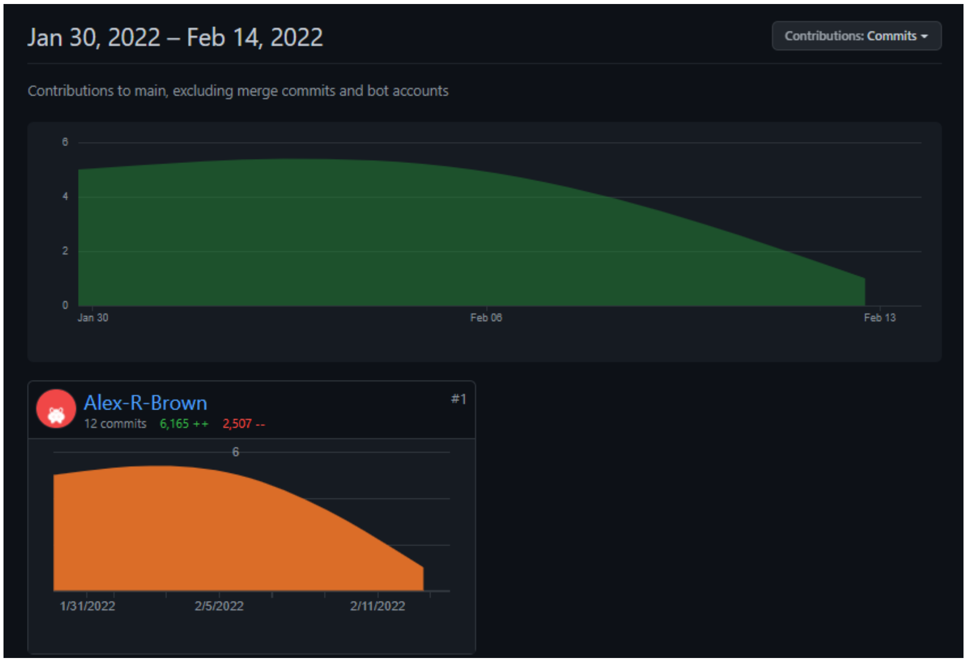
4. If you are behind schedule please provide a plan to address

- We are still communicating with our sponsor
- Keep regularly meeting
- Continue to meet with professors for insight

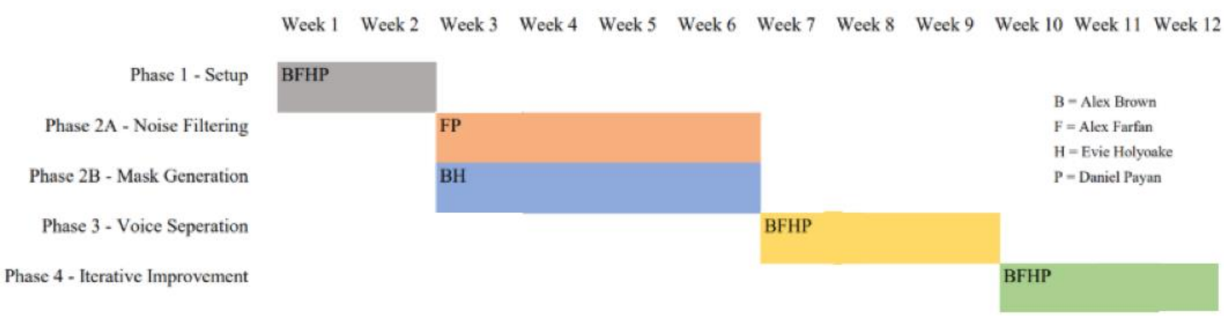
5. Describe your next step for the next 2 weeks

- Alex Brown & Evie Holyoake
    - Move focus to just voice separation
    - Research different ideas
        - Scaling/normalization
        - Decibel regulation/randomization
        - Reconstruction of preprocessing and MFCC vectors
        - Look into ConvTasNet's Model Diagram and using time series
    - More data
    - More in-depth data preparation
- Alex Farfan & Daniel Payan
    - Set up both a Notch and Match filter and determine which is more efficient
    - Use Coral Board to get microphone samples to determine file type and size

Git Repository:



Gantt Chart:



<u>Phase 1 - Setup:</u>

- Set up workspace/environment
  - Version control using Git
  - Python3 development using Jupyter Notebook through in an Anaconda environment
  - Simulink and Matlab development using Matlab IDE
- Begin implementing initial filters
- Use Matlab to set up FIR/CIC filters to familiarize ourselves with noise reduction implementations
- Set up/Initialize Coral Board
  - Will need a similar development environment for testing of system implementation

    - Produce .wav files using on-board Digital PDM Microphones
- Research further into quantitative analysis system performance
    - As this is an investigative development, standards of performance may be subject to change
    - Performance may be relative across iterations, rather than compared to a known metric, as MFCC mask voice separation is a not well-explored domain

<u>Phase 2A - Noise Filtering:</u>

- Alex Farfan and Daniel Payan will lead this phase
- Attempt to implement separation of background noise from voices
    - Separate background noise from a single voice
    - Proceed to separate background noise from multiple voices
    - Proceed to separate background noise from stacked voices
- Implement MFCC feature vector production through Matlab
- Output MFCC feature vectors to file (for later use by Python3 code)
- Begin testing filter performance by comparing clean audio samples to artificial noise that was filtered by the program

<u>Phase 2B - Mask Generation:</u>

- Alex Brown and Evie Holyoake will lead this phase
- Develop infrastructure for interfacing with potential audio clip databases
- Implement MFCC feature vector production through Python3 (utilizing the Librosa library)
- Input MFCC feature vectors from file (previously output by Matlab code)
- Research into mask generation based on existing in generating masks in the Time-Frequency spectrogram domain
- Setup system to create an “enrollment profile” of audio clips for individuals from database sources
- Visualize MFCC time-domain data
- Implement a mask generation machine learning algorithm
    - Uses MFCC feature vectors produced from enrollment profiles
    - Produces a “mask” of multipliers/scalars that represent an individual’s vocal presence in the MFCC-domain

<u>Phase 3 - Voice Separation:</u>

- Continue training mask generation model

- Use the MFCC feature masks to attempt to separate a known speaker from other speakers
    - Experimentally determine if MFCC feature mask multipliers are sufficient for attenuating non-target voices
    - Experimentally determine if a voice separation model is needed
- Reconstruct audio streams based on MFCC feature vectors with Python3 (utilizing the Librosa library)
- Implement the noise filtering algorithm on the Google Coral board, testing its performance
    - Experimentally determine how to use both microphones concurrently whether this improves performance
    - This must filter out noise recorded using the on-board microphone(s)
    - Determine how to improve performance to meet latency requirements

Phase 4 - Iterative Improvement:

- Iteratively improve noise filtering implementation
- Iteratively improve mask generation model
- Iteratively improve voice separation implementation
- Iteratively improve testing mechanics