Documentation: Word Recognition Web App

This project allows a user to record their voice and then see a computer interpret what they say. The user interacts with a webapp and the voice interpretation is done by machine learning. I am at the final stages of the first iteration of this project. At the moment, all units work and so they need to be connected to complete the app. There have been many implementation challenges up to this point including my lack of knowledge of javascript and html. There are also a few design challenges that persist; for the moment an inconsistent audio sample rate, audio recording length, and a small training set for the machine learning model prevent the first iteration of this product from predicting well. In the table at the end of this document I explore and visualize how the model’s prediction is sensitive to audio sample rate and the software used for recording ([audacity](https://github.com/Jakobovski/free-spoken-digit-dataset/blob/master/acquire_data/say_numbers_prompt.py), [recorder.js](https://addpipe.com/blog/using-recorder-js-to-capture-wav-audio-in-your-html5-web-site/))

Nonetheless I'm almost done with an end to end solution. Future feature improvements include: clipping the audio files only where there is sound (rn sound capture isn’t immediate), allowing user recorded data to be used as training points, and using librosa to compute the mfccs.

|  |  |  |  |
| --- | --- | --- | --- |
| Incorrectly Predicted | Incorrectly Predicted | Correctly Predicted: 1 | Correctly Predicted: 1 |
| 1\_alec.wav (recorded from web)  sample\_rate: 44100  frame\_length: 1102  frame\_step: 441  signal\_length: 15490  num\_frames: 34  pad\_signal\_length: 15655  frames: [[ -15.52 -17.12160297 -18.56695114 ... -71.78687575  -74.32695869 -75.36 ]  [-134.24 -138.3329511 -141.65303242 ... -210.71888948  -206.659348 -200.64 ]  [-368. -366.994359 -364.5364761 ... -26.4098874  -15.2014232 -3.92 ]  ...  [ 57.12 51.12478644 45.21692843 ... -87.87289808  -92.00861409 -94. ]  [ 68.4 73.28686131 80.10999178 ... -37.37399217  -39.44369282 -40.8 ]  [ 65.12 67.28629952 69.30594694 ... 0.  0. 0. ]]  mag\_frames: (34, 257)  pow\_frames: (34, 257)  high\_freq\_mel: 3923.337321740179  mel\_points: (42,)  hz\_points: (42,)  bin: (42,)  fbank: (40, 257)  filter\_banks: (34, 40)  dim1: 34  mfcc\_result: (63, 12) | 1\_alec\_4\_noise.wav (recorded in audacity @ 41kHz sample rate, then resampled, noise added) | 1\_alec\_4 (recorded in audacity @ 8kHz original sample rate) | 1\_yweweler.wav (original dataset sample)  sample\_rate: 8000  frame\_length: 200  frame\_step: 80  signal\_length: 2681  num\_frames: 33  pad\_signal\_length: 2760  frames: [[ 0.16 0.24068781 0.08091685 ... 2.42750542 2.24641953  1.92 ]  [ 0.56 0.24068781 -0.48550108 ... -0.48550108 -0.40114635  -1.76 ]  [-4.16 -2.40687807 -2.02292119 ... 6.95884888 7.06017567  6.96 ]  ...  [ 4.32 4.6532976 4.12675922 ... -0.40458424 -1.76504392  -2.4 ]  [-5.36 -5.61604883 -5.34051193 ... -0.48550108 -1.12320977  -0.72 ]  [-1.04 -0.80229269 -0.97100217 ... 0. 0.  0. ]]  mag\_frames: (33, 257)  pow\_frames: (33, 257)  high\_freq\_mel: 2146.06452750619  mel\_points: (42,)  hz\_points: (42,)  bin: (42,)  fbank: (40, 257)  filter\_banks: (33, 40)  dim1: 33  mfcc\_result: (63, 12) |