Accenter: Check In #2

Mark Lavrentyev (mlavrent)
Arvind Yalavarti (ayalava2)
Jeff Zhu (jzhu71)

Introduction

One major area of application of Deep Learning is speech recognition. While it has primarily had success, audio from non-native English speakers can negatively affect the accuracy of these systems. By debiasing accent from speech, these speech recognition algorithms can more effectively accommodate all speakers, regardless of accent.

Our project hopes to classify audio clips into various classes of accents. In other words, our algorithm will take in audio as input, and it will output one of several classes of accents. This would help contribute to better speech recognition algorithms, and possibly move towards better performing automated assistants such as Siri or Alexa.

If time permits, we hope to also translate audio clips between different accents, including a native speaker's accent. This will likely be a machine translation problem, which would move toward lowering barriers for communication between people with different accents.

Challenges

Current challenges during model development fall under two categories, processing hyperparameters and data transfer shape.

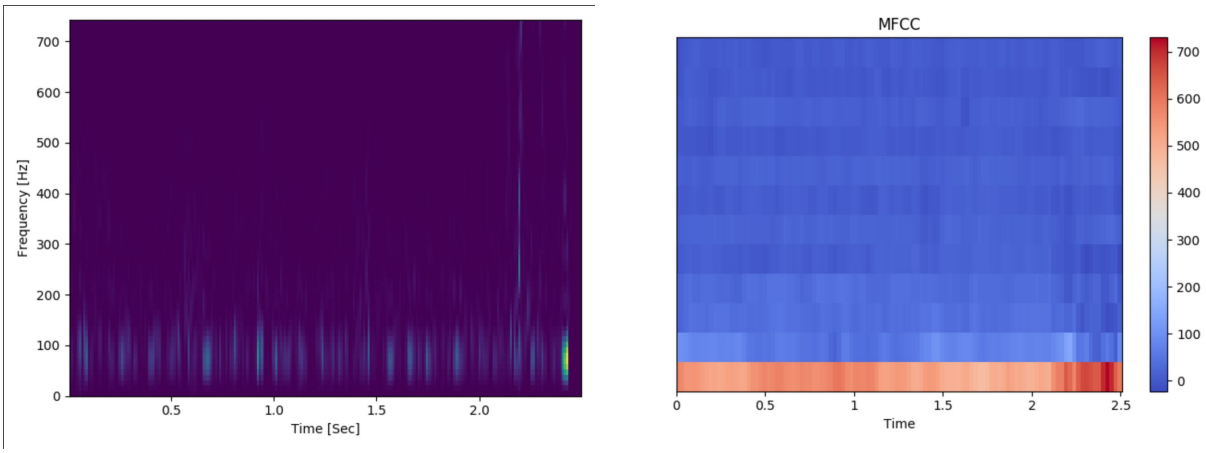
Segmenting Hyperparameters
When segmenting audio files based on white space, the silence length (minimum length of a silence to be used for a split) and the silence threshold (threshold below which audio is considered silence) needed to be selected based on the type of audio data being processed. These were hard coded as hyperparameters and might need to be updated during model training.

Data Transfer Shape
When sending and receiving Numpy arrays of segmented audio files and feature extracted audio files between different files and different phases of our pipeline, we needed to ensure that the shape of the arrays was consistent. For example, between segmentation and feature extraction, the shape of the data was of the shape [num_examples, audio_length].

Insights

We have finished developing our preprocessing and feature extraction libraries. The model construction is still in progress, so we do not have results regarding the model itself right now (but plan to by Friday). Our audio file segmentation and feature extraction (for both FFT and MFCC features) is complete and the results can be seen below:

The above image shows what we are doing to clip out silent audio from the audio clips. Red areas depict the sound clips that we keep, with the rest being cut out. We surmise that this will help our model learn to distinguish accents since it will not need to process silent data, which we believe is not important in distinguishing accents (i.e. the only important thing is the speech part). This will also help us filter out light background noise.



Above, we show the result of doing our two potential methods of feature extraction on a 2.5 second audio clip. This shows the transformation we are doing, into the frequency domain for FFT and into the speech feature domain for the MFCC extraction. We are using these since the literature we have consulted has consistently used some sort of feature extraction rather than having the model operate on raw audio wave files (which has shown poor results in the papers).

Plan

Our current plan is as follows:

- Finish data segmentation and feature extraction by Tuesday (11/26)
- Complete model architectures by Wednesday (11/27)
- Have a trained CNN model by Friday (11/29)
- Have a trained LSTM model by Sunday (12/1)

At this point, we will compare the two models and determine which works better for our domain. We will then go on to developing a model architecture for converting accents in speech.