Accenter

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.

Related Works

We are primarily relying on the following past works in developing our new model:

- Deep Learning for Classification of Speech Accents in Video Games
    - http://ceur-ws.org/Vol-2282/EXAG_114.pdf
    - Authors: Sergio Poo Hernandez, Vadim Bulitko, Shelby Carleton, Astrid Ensslin, Tejasvi Goorimoorthee
    - They achieved a 71% test accuracy over two classes. This paper converted the audio file into a spectrogram, and then used AlexNet to classify the data.
- Application of Convolutional Neural Networks in Accent Identification
    - http://www.andrew.cmu.edu/user/yyin1/resume/cnn_project_report.pdf
    - Authors: Keven Chionh, Maoyuan Song, Yue Yin
    - This paper achieved a training accuracy of 81.4% and a test accuracy of 77.9%. It trims/pads audio input to 30 seconds, extracts MFCC features, and then passes the features into a CNN.
- Accent Conversion Using Artificial Neural Networks
    - http://web.stanford.edu/class/cs224s/reports/Amy_Bearman.pdf

    - Authors: Amy Bearman, Kelsey Josund, Gawan Fiore
    - This work looks at a similar goal of using accent conversion to aid models whose primary goal is for speech comprehension by allowing the accent component of speech to be separated from the meaning component. They found that simple feed-forward networks work nearly as well as seq-to-seq LSTM models, something we hope to check.
- Deep Learning Approach to Accent Classification
    - http://cs229.stanford.edu/proj2017/final-reports/5244230.pdf
    - Authors: Leon Mak An Sheng, Mok Wei Xiong Edmund
    - This paper extracted time windows of 0.18 seconds. They first used the Librosa library to extract MFCCs. Then, they compared the performance of using an MLP and a CNN. The latter performed better. They outperformed traditional machine learning methods such as gradient boosting and random forest.
- Accent Classification of Non-Native English Speakers
    - http://web.stanford.edu/class/cs224s/reports/Albert_Chu.pdf
    - Albert Chu, Peter Lai, Diana Le
    - This paper did not produce a good accuracy. However, they experimented with performing PCA after extracting MFCC features. Also, it compared using an LSTM vs. a CNN.

Data

Our primary dataset is the Wildcat Corpus of Native- and Foreign-Accented English (Wildcat Corpus) collected by the Linguistics Group at Northwestern University. This data consists of pairs of Native+Native and NonNative+NonNative recordings of scripted and unscripted audio separated into lists of clearly enunciated English words by native language speakers from different regions.

The dataset consists of 8 pairs of English audio, 5 pairs of Mandarin audio, and 4 pairs of Korean audio. For the first iteration of our classification task, we plan to use these 3 nationalities as our classification classes. If time and logistics permit, we plan to augment our dataset with accent data from British-English and Australian-English audio.

We plan on extracting audio signals from the raw transcripts where words were uttered. This would involve a peak detection algorithm where we search for windows in the raw audio data where the energy density (defined as squared signal amplitude) is greater than the average energy density of the entire file. This would allow us to train the model on individual words/a sequence of words.

After running the initial preprocessing algorithm, we plan to experiment between two forms of feature extraction. The first of these would be a simple Fast Fourier transform (FFT) which would map the signal data into the frequency domain, preserving time and amplitude. The next would involve generating the Mel-frequency cepstral coefficients (MFCC), a feature of audio signals that are used for identifying linguistic content.

Methodology

We plan to run two main experiments (and potentially a third, time permitting). We want to experiment with two preprocessing methods for our classifier network. The first method we will look at will be simply lifting the raw audio files into the frequency domain using a Fast Fourier Transform (FFT). We hope that this will allow our classification model to operate on frequency-domain data, which may be easier to learn patterns from than raw audio wave data. Our other method that we will test is using Mel-frequency cepstrum coefficients (MFCC). This is a more involved preprocessing method that picks out features similar to how humans perceive speech, which may give our model more of a boost in picking out patterns for each accent among the more processed features. Provided we have time, we will also test potentially using Principal Component Analysis (PCA) after applying MFCC to limit the inputted features to only the most important ones.

Our second experiment will involve testing two architectures for classifying accents from speech data: a CNN and a LSTM. The CNN model will take as input an MFCC processed audio clip of some time length. The result of an MFCC looks similar to an image, which allows us to then apply a CNN to it. Our baseline architecture will use five convolutional layers, with filters gradually decreasing in size, each separated by pooling layers. This will be followed by four fully-connected layers. We will adjust the architecture depending on whether the model is able to learn the classifications correctly. We will also experiment with using an LSTM, which will be applied to a sequence of audio clips (each MFCC processed) to produce a classification from its hidden state (which is then run through several dense layers).

Provided we have time, we will also attempt to create a model for accent conversion (i.e. taking speech in one accent and converting it to speech in another accent). One potential architecture for this will be a LSTM seq-to-seq model that first processes each time window's MFCC output using a CNN, encodes it, then decodes it, producing frequency-domain data that can then be turned into audio data using an inverse FFT. We also hope to try this same model using a Transformer model instead.

Metrics

For our classification model, we plan to use simple accuracy for quantifying how well our trained model does in classifying accents. This is simple to implement and is a good metric as it tells how well the model does overall. We also plan to split out the accuracy per accent category to check that the model performs roughly equally well on all our categories and does not do significantly worse on a small minority of the data.

For the accent transfer model, we plan to use our classifier to distinguish the source and target accents. Using the source accent probabilities as a baseline, the target accent probabilities will be scaled and then used as a metric for how well the transfer model learned to convert from source accent to target accent. To verify, we will also do qualitative checks by listening to the results of the transfer model and comparing to real examples of speech in the target accent.

Ethics

We chose this problem because it has potentially broad applications in helping reduce bias in existing audio-to-text models. In particular, automated assistants such as Siri and Alexa rely entirely on recognizing what people are saying and converting that to text before actually trying to compute the semantics of what the speaker said. We hope that our work may have potential applications in helping debias these models with respect to accents (which can be used to infer race, nationality, etc.). Classification is a start to this goal, while our potential reach goal is to do accent transfer, which would allow for full debiasing of accents.

We are planning to use two datasets as mentioned above - the Wildcat Corpus provided by Northwestern and the IDEA dataset. Both are collected by respected members of the linguistics and NLP communities (many are professors at universities). The Wildcat Corpus does have the issue that it is biased towards Asian speakers of English (e.g. Chinese, Korean, etc.). This may present a problem when we try to classify other accents, such as those of native speakers of various European, African, and other languages. However, we hope to supplement this with the IDEA dataset, which is fairly comprehensive (and also notes the regions where it is less complete) so that we can predict other English accents as well. We have chosen to classify accents based on the nationality of the language as this is how most of our data is split up. This may be somewhat problematic as it assumes that people in a single nation have similar English accents, even though this may not be actually true, as accents vary by region in many countries.

We plan for our work to be used in debiasing accents such that other models can make use of underlying speech data without picking up on patterns that may cause the model to learn some discriminatory features. Errors in our model would have fairly small effects in the real world, since a model that uses this debiasing but fails to fully debias correctly will likely still work correctly, although it may begin to pick up on discriminatory patterns, which could be a problem.

Division of Labor

We have divided this project into the following parts: data sourcing and basic preprocessing, MFCC vs. FFT data processing experiment, CNN vs. LSTM for accent classification experiment, feedforward network vs. LSTM seq-to-seq experiment for accent transfer experiment.

Arvind is responsible for data sourcing and basic preprocessing. This will include connecting with the owners of datasets as well as writing code to consume raw audio (wav) files and process them using either MFCC or FFT. Jeff is responsible for overseeing the MFCC vs. FFT experiment to determine which preprocessing technique allows our classifier network to achieve the best test accuracy. Mark is responsible for running the CNN vs. LSTM experiment for accent classification to determine which architecture will work best for classifying accents based on speech data.

Provided we have success with our accent classification experiment and have time to proceed to the accent transfer experiment, we will split the work such that each member handles developing and training one of the architectures given in the Methodology section.