Abstract

Automatic speech recognition (ASR) systems would ideally be able to accurately capture speech regardless of the speaker. However, accent is often a confounding factor, and having separate speech-to-text models for each accent is less desirable than a single model. In this paper we propose a methodology for accent conversion that learns differences between a pair of accents and produces a series of transformation matrices that can be applied to extracted Mel Frequency Cepstral Coefficients. This is accomplished with a feedforward artificial neural network, accompanied by alignment preprocessing, and validated with MCD and a softmax classifier. Results show that this approach may be a useful preprocessing step for ASR systems.

1 Introduction

Among the many issues facing Automatic Speech Recognition (ASR) systems, effectively handling accents is one of the most challenging. Particularly when working with languages that have highly varied pronunciations, such as Spanish, English, and Chinese [17], an ASR system trained on only one accent might only be effective for a minority of the speakers of that language. This does not include non-native speakers who learn a language and carry over their native accent, a population that expands the need for proper handling of accent variation. Frequently, ASR systems perform much better for users with the same accent as the training data used to develop the system. This is due to the way accents affect prosody, enunciation, vowel sounds, and other aspects of speech, which in turn change the resulting MFCC or other features that are used for speech recognition. We propose a system to transform speech from one accent to another as a way of addressing this problem.

In particular, we propose applying a simple feedforward neural network with various preprocessing steps to learn a series of conversion weight matrices between a source and target accent. The resultant trained matrices accept MFCCs representing an utterance in one accent and output MFCCs for the same utterance in a different accent. We evaluated our model with both the mel-cepstral distortion measure of MFCC difference and a neural classifier to detect the degree to which our result sequences of MFCCs truly resemble the desired accent.

Theoretically, an ASR system could be implemented with a separate model for each anticipated accent. Compared to our approach, this would require the same number of trained models (one for each accent). However, our approach requires significantly less training data, and thus less training time, because it breaks out the accent portion of the overall speech recognition problem, avoiding duplication of the rest of the training necessary for speech recognition.

This issue of understanding varied accents arises in most languages, and, accordingly, the approach we use could be applied to any language. However, for simplicity of development and due to available training data, we trained and tested our model on the English language with American, Indian, and Scottish accents, for both genders.

2 Background and Related Work

Voice conversion is an active area of research, but the majority of papers on the subject focus on modifying the voice itself, not the pronunciation. [9], [10], and [11] demonstrate that it is possi-
ble to reconstruct a speech sound from mel frequency cepstral coefficients, although it typically requires additional inputs for accurate reconstruction. [10] used a pitch excitation signal in concert with MFCCs as an input into a source-filter model which resulted in more natural-sounding speech. [11] similarly used pitch data, but they instead derived sine-wave frequencies from the pitch and used this to invert the original binning step in MFCC computation. [15] compared performance of Gaussian Mixture Models to DNNs map of spectral features of a source speaker to that of a target speaker, converting the speaking voice while maintaining the content of speech. They used f0 transformation for both models and optimized mean squared error of transformed MFCCs in the neural network and found that the best results were obtained with a four-hidden-layer neural network with hidden layers of variable size.

[14] applied Convolutional Neural Networks to the same problem in an attempt to modify not just pitch but also timbre, with the intent of improving the similarity between the target speaker’s voice and the generated voice. They both transformed speech directly and built generative models to sound like a particular person through use of generative adversarial networks and visual analogy construction. [13] employed deep autoencoders to train in a speaker-independent fashion, which allowed them to build representations of speaker-specific short-term spectra. They ultimately modified input voices to match some target voice and performed both objective (reconstruction error) and subjective (human perception) evaluations. All three of these neural-net-based voice modification projects have a similar intuition to what we propose, but they do not deal with accents broadly but rather more specifically individuals.

Neural networks (particularly deep neural networks) have been shown to be particularly effective for representing sequential information such as language, video, and speech. Generally, DNNs also serve as accurate classifiers. [12] used both DNNs and RNNs together in a single classifier to identify accents, with the DNN focused on longer term statistical features and the RNN on shorter term acoustic features. They found that this system outperformed either DNNs or RNNs when used alone. [16] created an audio generation model using hierarchical RNNs that consisted of different modules focused on learning audio variations over different time spans, with the goal of capturing both short term features and long term dependencies. A 3-tiered flavor of this approach was greatly preferred by AB test subjects over both an unconditional RNN solution and a WaveNet implementation.

We take particular inspiration from [16], which compared a GMM model to an ANN model to convert a female voice to a male one with the same utterance. This study used the same dataset we have access to, making their results particularly relevant to us. They also showed that a remarkably simple model can perform very well in this problem space. One further major area of related work is in accent classification. This is the problem of inferring the native language or regional identity of a speaker from his or her accented speech. Similar features that allow for accent identification/classification are relevant for accent conversion since the aspects of an accent that characterize it are precisely what must be changed for conversion. Further, to identify whether an accent has been successfully reconstructed after a conversion process, a classifier is very useful.

Spectral features and temporal features such as intonation and durations vary with accent. These features have been used in statistical models such as Gaussian Mixture Models (GMMs) and Hidden Markov Models (HMMs) to discriminate between several different accents. [3] used GMMs trained with formant frequency features to discriminate between American English and Indian accented English. [2] identified Flemish regional accents by providing formant and phoneme duration features as input to the eigenvoice method [5], which is a dimensionality reduction technique for speaker models. [6] proposes a linear discriminant analysis (LDA) approach (essentially, a form of dimensionality reduction) on individual phoneme classes and extended to continuous speech utterances, in order to classify three different types of accents. [7] and [8] used support vector machines (SVMs); [7] trained SVMs with MFCC features and [8] trained using word-final stop closure duration, word duration, intonation features, and the F2-F3 contour which captures tongue movements.

3 Approach

We used parallel utterances from American, Indian, and Scottish English, extracted MFCCs, aligned them using fast dynamic time warping
(FastDTW), and fed the resultant features through a feedforward neural network to learn conversion weight matrices.

3.1 Dataset

The CMU Arctic dataset consists of 1150 samples of text spoken by men with American, Canadian, Scottish, and Indian accents, and a woman with an American accent. Since the American and Canadian accents sounded nearly identical to our ears, we used only the American accent for this project. We extracted 25 mel cepstral coefficients from each 5ms frame with 100 frequency bands in each of the training samples and paired samples of identical utterances in two different accents for the source and target data into our system. Each feature vector was zero-padded or truncated to the same length, which we set to be 1220 frames per sample.

3.2 Alignment

After extracting the MFCCs, the source and target were aligned using FastDTW. This is an O(N) time approximate alignment algorithm that minimizes squared error between the two samples. Alignment is necessary because people speak at different rates and without alignment it is much harder for the system to identify which differences are due to accent and which are due to rate of speech.

3.3 Artificial Neural Network

We constructed a feedforward neural network with two hidden tanh layers of size 100 and a final linear output layer. Both the input layer and output layer were of size 25, since we used 25 coefficients from each 5-millisecond time period. The model learned the weight matrices for the two hidden layers and the output layers, which started with xavier initialization, and we found that performance was much better when trained without biases. Figure 1 explains this in more detail, including pre-and post-processing steps.

Specifically, our model involved the following computations on the prediction step:

\[ z = \text{input}_\text{MFCCs} \cdot W_1 \]
\[ h_1 = \tanh(z) \]
\[ z = h_1 \cdot W_2 \]
\[ h_2 = \tanh(z) \]
\[ \text{predicted}_\text{MFCCs} = h_2 \cdot W_3 \]

Note the lack of a nonlinearity on the final prediction layer. All weights are learned for all timesteps in the data simultaneously, allowing the lack of temporal awareness by the feedforward architecture to not be a handicap in learning.

3.4 Waveform Reconstruction

After predicting MFCCs for the target accent, we reconstructed the waveform using a MatLab implementation of InvMFCC. This is a lossy function, as MFCCs do not retain all information about speech sounds that are perceivable, so the resultant waveforms were guttural and noisy. Pitch information in particular is lost in the MFCC transformation.

4 Experiments

4.1 ANN Model

4.1.1 Architecture

Our final model used Adam optimization to minimize mean squared error over 5,000 epochs with
batch size 16. We first tried basic gradient descent, then noted that papers frequently made use of momentum for similar tasks and used TensorFlow’s MomentumOptimizer before trying Adam optimization. After experimentation with various learning rates, batch sizes, numbers of epochs, and momentum values, we found that similar hyperparameters worked for all three of our dataset pairs (US-Scottish, US-Indian, US female-US male).

We evaluated our model, as in [16], with Mel Cepstral Distortion, which is a weighted average of squared differences between two sets of mel frequency cepstral coefficients attuned to the perception of the human ear.

\[
MCD = \frac{10}{\ln 10} \sqrt{2 \sum_{i=0}^{24} (mc_1^{(i)} - mc_2^{(i)})^2}
\]

### 4.1.2 Classifier

To evaluate our model’s performance, we created a softmax classifier to predict an accent label from MFCC data parsed identically to the parsing in our primary conversion model. This took the form of a feedforward ANN with two hidden tanh layers and a softmax output, with hidden sizes 750 and 1000 and cross entropy loss. The classifier achieved 92.9% accuracy in binary classification on the benchmark American English versus Scottish English task, significantly outperforming the 68% accuracy of a Naive Bayes classifier and 76% accuracy of a Support Vector Machine classifier for the same problem.

### 4.1.3 Results

Our model achieved MCDs below 10 for all three of the conversions we attempted. The state of the art for voice gender conversion is 6.9, which we were able to approach; there is no benchmark for accent conversion, but our MCD scores are quite close for that task as well.

<table>
<thead>
<tr>
<th>Accents</th>
<th>Train MCD</th>
<th>Val MCD</th>
</tr>
</thead>
<tbody>
<tr>
<td>US to Scottish</td>
<td>9.67</td>
<td>9.84</td>
</tr>
<tr>
<td>US to Indian</td>
<td>8.93</td>
<td>8.93</td>
</tr>
<tr>
<td>US female to male</td>
<td>8.16</td>
<td>8.17</td>
</tr>
</tbody>
</table>

Table 3: MCD Results

Figure 2 shows the similarity between the frequencies of predicted and target utterances and Figure 3 demonstrates the same comparison for the waveforms. Frequencies and waveforms for these plots were both computed after MFCC computation and conversion back to a wave file for both target and prediction to eliminate disparities due to the lossy nature of MFCC calculation. The differences between the prediction and target are visible, but the general shape for both the frequencies and the waveform are similar between the two.
4.2 Other Methods Tried

4.2.1 Sequence-to-Sequence LSTM-RNN

The first method we used to approach this problem was a sequence-to-sequence LSTM-RNN, building off of the intuition of neural machine translation. We hoped to learn a statistical representation of each accent which could then be used to generate the same utterance in a new accent. This would have the benefit of taking advantage of temporal information in the utterances that is lost in a feedforward architecture. Initial results were no more promising than the simpler feedforward model, however, and we had more literature to back up focusing on that model for this particular problem.

4.2.2 Denoising Autoencoder

Denoising autoencoders (DAEs) are unsupervised models that learn how to reconstruct their input and remove some added noise at the same time. They consist of an encoding step and a decoding step which operate on the same learned weight matrices and bias vectors. We hoped to learn two DAEs, one for the source and one for the target accents, and then use the learned weight matrices for each of these to encode one accent and decode it into the other. Our DAE successfully denoised each of the input accents back into itself, but was less useful for accent modification. This could be a good avenue for future research.

4.2.3 Post-MFCC-Reconstruction Improvement

All of our three attempted model architectures learned best with MFCC features, but the MFCC and inverse MFCC process is very lossy so reconstructed sound files do not sound natural. We therefore built a postprocessing model with similar intuition and architecture to our most successful feedforward ANN that, rather than learning to convert one accent to another, learned to convert a waveform that resulted from the MFCC-InvMFCC process to the original waveform. We created training data by computing the MFCCs and then inverting them for all sound files in our original CMU Arctic dataset, then trained by pairing the MFCC-only file as input with its original as the target. The goal was to have the model learn restorative transformation matrices that would negate the observed degradation patterns of MFCC-InvMFCC conversions and then apply those transformation matrices to the waveform output of our accent conversion model. While this showed modest success in subjective sound quality, it was not quantifiable.

4.2.4 Alternative Features

Given the poor reconstruction abilities of MFCCs, we also experimented with training on raw wave files and on Fourier Transform features. Using just 1/16000-second-long samples from the raw waveform was the simplest method tried since it required no processing or reconstruction at the end, but it performed poorly since alignment has little meaning on a vector of this form and there is too much variation to learn. The Fast Fourier Transform algorithm is quick and fully invertible via the Inverse Fast Fourier Transform, which is an attractive quality since we need to revert back to a waveform from a feature vector. Models trained on Fourier Transform data performed better than MFCC-based models after a few epochs, but then ceased to continue to learn.

5 Conclusion and Future Work

The feedforward architecture successfully converts the MFCCs of a sample from one accent to another, but loses other speech characteristics that are not represented by MFCCs. Future work should focus on integrating other features into the model to use in reconstruction, perhaps starting with rescaling the wavefiles reconstructed from the predicted MFCCs using pitch data of some kind. Alternatively, the waveform degradation problem might be solved if similarly successful accent conversion could be achieved with less lossy features than MFCCs.

While the results of the simple feedforward model are gratifying, more complex models should be able to capture additional information about utterances and accents that this model does not. The intuition behind denoising autoencoders seems extremely relevant to this problem space, suggesting that there is some implementation that would lead to greater success. Particularly, learning with additional or alternative features besides
MFCCs may be more successful with such architectures. The ability of RNNs to capture temporal information should also be further explored, as such information is certainly relevant to the differences between accents.

As discussed in the introduction, however, one of the primary uses of a system such as this would be as an initial processing step in a speech recognition system. In that case, the poor reconstruction of the wave file may not matter; all that would be required would be accurately predicting the features used by the rest of the system. In that case, additional hyperparameter tuning or additional data acquisition would be useful to drive the MCD score lower, indicating even more faithful accent conversion.

6 References


[10] B. Milner, X. Shao, Speech reconstruction from mel-frequency cepstral coefficients using a source-filter model, School of Information Systems, University of East Anglia, Norwich, UK.


[12] Yishan Jiao, Ming Tu, and Julie Liss, Accent Identification by Combining Deep Neural Networks and Recurrent Neural Networks Trained on LSTM, Arizona State University.

[13] Seyed Hamidreza Mohammadi and Alexander Kain, Voice Conversion Using Deep Neural Networks with Speaker-Independent Pre-Training, Center for Spoken Language Understanding, Oregon Health & Science University, IEEE.

