Speech Accent Classification

Corey Shih
ctshih@stanford.edu

1. Introduction

English is one of the most prevalent languages in the world, and is the one most commonly used for communication between native speakers of different languages. As such, people from different regions around the world exhibit unique accents when speaking English. Classifying these accents can provide information about a speaker's nationality and heritage to speech recognition systems, which are becoming increasingly common in day-to-day life. The data gleaned from a speaker's accent can help speech recognition systems identify topics more relevant to the user, for the purposes of search results or advertisements.

This project attempts to classify amongst 4 common accents (British, French, Spanish, and Mandarin) from audio samples of accented speakers speaking English. The 13^{th} lowest order mel-frequency cepstral coefficients (MFCCs) of the audio signals are used as inputs to the algorithms. A softmax regression model and a long short-term memory (LSTM) neural network are used to predict the accent of the speaker from each audio sample.

2. Related Work

The most successful previous work in this area utilizes a dictionary of words known to be sensitive to foreign accents and develops individual word and phenome based classification algorithms, using MFCCs as features.[1][2] In doing so, a classification accuracy of 93% among 4 different accents is achieved. Unfortunately, I do not have access to such an extensive database, and hence cannot replicate such results. Instead, I attempt to classify accents directly from the MFCCs of each sample.

In a more recent paper, Choueiter et al. attempt 23-way accent classification using heteroscedastic linear discriminant analysis and obtain a classification accuracy of 32%.[3] Such an approach is more similar to mine, as a dictionary of accented words is not constructed beforehand. However, the accuracy of the algorithm clearly suffers.

3. Dataset and Features

Audio samples were taken from the Speech Accent Archive, which provides clips of different accented speakers reciting the same English paragraph, as well as information about their geographical location, gender, and native language.[4] Samples were taken from 4 of the accent categories with the most examples (British, French, Spanish, and Mandarin) for a total of 430 examples, split into 386 training examples and 44 test examples. As different speakers speak at different rates, the audio signals were resampled to be roughly the same length ($\sim 10^6$ length vectors) to better match up with each other.

MFCCs are commonly used features in speech recognition systems because they approximate the important features that the human auditory system detects in audio signals. MFCCs are obtained by taking the Fourier transform of the audio signal, mapping the spectrum powers to the mel scale, and then taking the discrete cosine transform of the logarithms of the powers. The MFCCs

represent the amplitudes of the resulting spectrum. Using Ellis' MFCC toolbox for MATLAB, the 13th lowest order MFCCs of each resampled audio example were extracted using a frame time of 25 ms and a frame shift of 10 ms.[5] As the resulting matrices were extremely large and caused training times to be exceptionally long, the features were trimmed to 13 × 500 matrices for each example, as this seemed to result in the best compromise between model performance and training time. For use in the softmax regression model, the matrices were flattened to 1 × 6500 feature vectors. An example of the MFCC feature matrix is shown in Figure 1.

Figure 1. MFCCs for English Training Example 1.

4. Methods

Two models were constructed for this project: a softmax regression model and a LSTM network. The softmax regression was used as a baseline to evaluate the effectiveness of the LSTM model.

Softmax Regression. Softmax regression is commonly used for classification of multinomial data. For each of the k values the response variable y can take on, the conditional distribution of y given the features x and parameters θ are calculated by

$$p(y = i|x;\theta) = \frac{e^{\theta_i^T x}}{\sum_{j=1}^{k} e^{\theta_j^T x}}$$

The value of i with the highest conditional probability becomes the output of the model.

LSTM. LSTMs are a type of recurrent neural network capable of learning long-term temporal dependencies in data. This makes them naturally suited to classifying sequences. LSTMs are already commonly used in natural language processing and automatic speech recognition, making them a natural choice for accent classification.

The neurons in a hidden layer of a recurrent neural network form a directed cycle, allowing them to pass information to each other and exhibit temporal behavior. Additionally, in LSTMs, a forget gate allows the LSTM cells to either remember or forget their previous state, allowing for the establishment of long term dependencies. An illustration of a typical LSTM cell is shown in Figure 2. The input, output, and forget gates of a LSTM typically use sigmoidal or hyperbolic tangent activation functions.

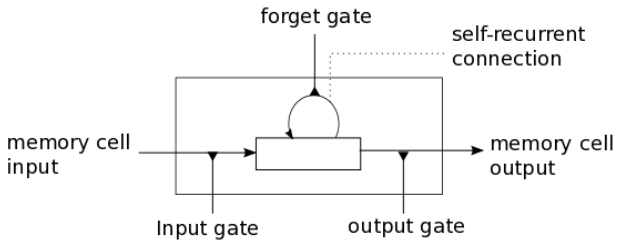

Figure 2. Schematic of LSTM cell.[6]

For this project, I used MATLAB's Neural Network Toolbox to construct a neural network with a single LSTM layer and used a softmax activation function for the output node.[7] Mini-batch gradient descent with momentum was used to optimize the parameters, with a learning rate of 0.01 and a mini batch size of 30. The learning rate and mini batch size were chosen based on resulting model accuracy

and training time. Models were trained for a maximum of 100 epochs, and L2 regularization was utilized with a value of 0.001 for γ.

5. Results and Discussion

The performance of each model was based on accuracy of the model over the training and test sets. Training and test accuracies for both models are given in Table 1.

Table 1. Training and test accuracies for softmax and LSTM models.

Model	Training Accuracy	Test Accuracy
Softmax	54.40%	38.64%
LSTM	79.02%	52.27%

Confusion matrices for both models are given in Figures 3 and 4. The output classes 1, 2, 3, and 4 correspond to British, Spanish, French, and Mandarin accents, respectively.

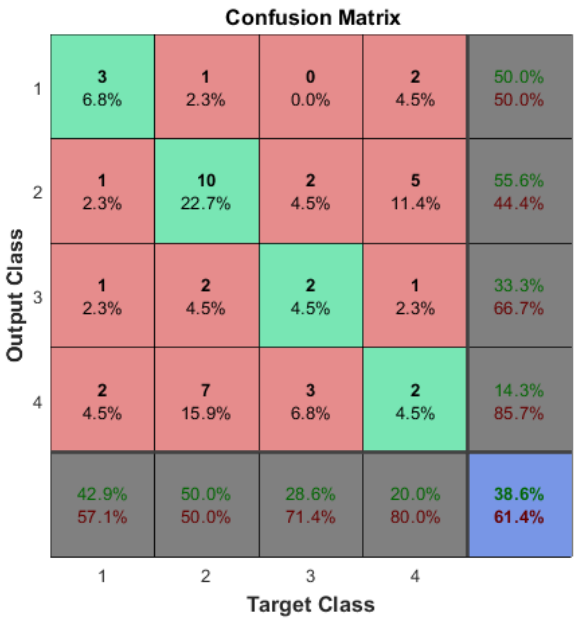

Figure 3. Confusion matrix for softmax model.

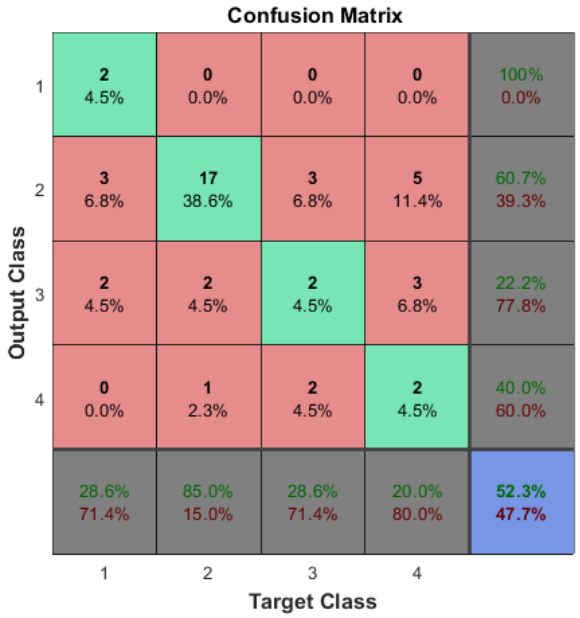

Figure 4. Confusion matrix for LSTM model.

From Table 1, it is clear that both models achieve higher accuracy on the test set than random guessing among the 4 categories (25%), and the LSTM model performs significantly better than the softmax model, though the resulting accuracy still leaves much to be desired. The better performance of the LSTM model comes as no surprise, as the algorithm is specifically designed to classify time-dependent sequences. Comparing the training and test accuracies suggests that both models were overfit to the training set despite the L2 regularization used; higher values of γ or additional regularization techniques such as dropout or cross validation may be required to achieve more similar values for the training and test accuracy.

Figures 3 and 4 reveal that both the softmax and LSTM models perform significantly better on Spanish accents than on any other accents. For the LSTM model in particular, the classification accuracy for Spanish accents was 85%, compared to accuracies between 20-30% for all other accents. This means that the increase in the overall 52.27% test accuracy for the LSTM model over the base of

25% is almost entirely due to the model's aptitude for correctly classifying Spanish accents. This is likely due to the fact that there were many more Spanish accent examples in the dataset than examples from other accent types. Of the 440 total examples, 199 were Spanish, whereas 68 were English, 67 were French, and 96 were Mandarin. If there were more examples for other accent types in the dataset, the performance of the model on the other accents would improve.

The less-than-ideal performance of the models could in part be due to irregularities in the dataset. Upon inspection of individual audio samples in the dataset, it became clear that not all speakers under a certain accent category in the Speech Accent Archive actually exhibited an accent. While the Speech Accent Archive categorizes audio samples based on the native language and geographical location of the speaker, not all of these speakers had a strong accent, if any accent at all. The prevalence of non-accented speakers in the training and test sets would degrade the performance of the models; I would have to manually sift through the dataset and remove any samples I thought did not exhibit a strong accent.

The dataset used for this project was extremely small when compared to the size of datasets used in most machine learning algorithms. I only had a few hundred examples total split over 4 categories, while many machine learning algorithms utilize datasets with sizes in the tens of thousands. The incredibly small size of the dataset I used negatively impacts the performance of the models trained, as there are simply not enough training examples to obtain accurate values for the parameters. Indeed, the accent category that the models performed best on was the category with the largest number of examples. Finding a larger dataset to train on is imperative for improving model performance.

6. Conclusion and Future Work

Using the MFCCs of audio samples obtained from the Speech Accent Archive, I trained a softmax regression and LSTM neural network model to classify amongst British, Spanish, French and Mandarin accents. The LSTM model performed significantly better than the softmax regression model, achieving a test accuracy of 52.27%, compared to the softmax model's test accuracy of 38.67%. Both models performed better on Spanish accents than on the other accents, most likely due to the much larger number of Spanish examples in the dataset. Model performance was degraded due to the low overall number of training examples and the presence of non-accented speakers in the various accent categories.

A number of additional methods could be utilized to improve the performance of the models trained in this project. Dynamic time warping could be used to sync the audio signals more effectively than simply resampling them. Alternatively, the audio samples could be split into individual words and fed into the algorithms as separate features. Furthermore, an accent classifier would ideally be able to classify accents regardless of the actual English words being spoken, but the dataset used for this project has all speakers reciting the same English paragraph. Obtaining a much larger dataset of accented speakers speaking various English phrases would be necessary to build a more general accent classifier.

7. Contributions

All work on this project was done by Corey Shih.

8. References

1) Arslan, L.M.; Hansen, J.H.L. Language accent classification in American English, *Speech Commun.* **1996,** *18,* 353-367.
2) Arslan, L.M.; Hansen, J.H.L. Foreign accent classification using source generator based prosodic features, in *ICASSP-95,* **1995,** *Detroit, MI, USA.*
3) Choueiter, G.; Zweig, G.; Nguyen, P. An empirical study of automatic accent classification, in *ICASSP 2008,* **2008,** *Las Vegas, NV, USA.*
4) Weinberger, S.H. Speech Accent Archive, **2015,** http://accent.gmu.edu
5) Ellis, D. PLP and RASTA (and MFCC, and inversion) in Matlab using melfcc.m and invmelfcc.m, **2012,** http://labrosa.ee.columbia.edu/matlab/rastamat
6) Carrier, P.L.; Cho, K. LSTM Networks for Sentiment Analysis, **2017,** http://deeplearning.net/tutorial/lstm.html
7) MathWorks. Neural Network Toolbox 11.0, **2017.**