UCLA
Dept. of Electrical and Computer Engineering
ECE M214A: Digital Speech Processing
Fall 2024

Noise Robust Speaker Region Identification

Project Description

Oral presentations and code submission are scheduled for **Wednesday, December 4th** during class time (**2 pm - 4 pm**). For MSOL students, it is scheduled for **Saturday, December 7th** at **12-2 pm** by Zoom.

I. Introduction

In this project, we are interested in implementing a speaker region identification system that predicts the city of origin of the speaker of a given utterance. Acoustic features such as MFCCs, LPCs (and others) may be used to perform the task. The challenges of this task arise both due to the limited training data available, and the mismatch between the training and test conditions.

II. Data

The Corpus of Regional African American Language (CORAAL) [1] contains speakers each belonging to one of six different US cities: 1) Rochester, NY (ROC), 2) Lower East Side, Manhattan, NY (LES), 3) Washington DC (DCB), 4) Princeville, NC (PRV), 5) Valdosta, GA (VLD) or 6) Detroit, MI (DTA). We selected a subset of speakers from the CORAAL dataset, and only the utterances in the corpus with length greater than 5 seconds.

Your model will be tested on a blind test set. The blind test set will consist of a different set of speakers from the above cities. All the wav files provided to you were originally sampled at 44.1kHz.

III. Project Codebase

The enclosed project link provides access to the following folders with utterances: train_clean, test_clean, test_noisy. That is, the test utterances are in one dataset “clean" and in the other, a noise masker (additive) was added (babble noise added with an SNR of 10 dB). The following python notebook [2] contains functions that extract features from the utterances in the above folders, and calculates the accuracy through the use of a xgboost based classifier.

Details: The baseline system consists of the following: MFCC features (13) extracted from the utterances and used for training the xgboost model. More detailed instructions on the specific libraries used for feature extraction and the model are present within the python notebook.

Note: If you modify the classifier and/or use a different classifier, report results with the

baseline classifier and the modifications

IV. Objectives

Your task is to derive a set of features and implement them to predict the city of origin of the speaker of every utterance. You will train on clean data and test with 1) clean data, and 2) noisy data. You can run the script either on google colab, or on a personal laptop.

Note: Do not use the noisy data for training

V. Evaluation Metrics

Evaluation of the project will be primarily based on the performance of the trained classifier. The accuracy calculated by the python script will be used for evaluation. Along with the Accuracy, report other classification metrics, such as the Confusion matrix of your model. Results on both clean and noisy data, as well as performance on a blind test set, will be considered for evaluation.

In addition to performance on the test sets, the explainability of the success of the model is also considered. In this project we provide code for conducting Shap analysis, which can help quantify the impact of individual features in the classifier, and the reasoning behind their effectiveness. Experimenting with a diverse range of features is encouraged.

VI. Instructions

A. Setting up the project

a. Download project package from Box [3].
b. Unzip the compressed file.
c. Upload the 'project_data' folder to your Google account
d. Make a copy of the colab notebook [2]
e. Open the notebook and run all the cells

B. Run custom features from Python

a. In the accompanying python notebook, edit the extract_features function to calculate your custom features directly
b. Run the model training and inference steps directly.
c. Repeat for other trials.

VII. Baseline Results

The baseline script using MFCCs should take around 30 mins to run on Google Colab.

Dataset	**Accuracy**
Test Clean	**50.65%**

Test Noisy	**43.79%**

VII. Oral Presentations

There will be oral presentations by the different teams describing their work. Presentations should be planned by the team as a group.

VIII. Report and Code

The report (one per group) should include:

- Introduction (what is the problem/why is it important)
- Background (literature survey)
- Project Description (features, algorithm, implementation, results, average run times, etc.)
- Summary and Discussion (also ideas for future work)
- References (cited throughout the report)
- Figures and flowcharts generally help clarify the text.

The report should be 4-pages long (excluding references) and have the same format as the INTERSPEECH conference.

The code should be turned in on the day of the presentation. Comments at the beginning of each function should describe what the function intends to do.

To evaluate the robustness of your system, we will use speech from a different set of unseen speakers to evaluate the system performance. We will run your scripts on the unknown data and submit the scores. The final report may be turned in on **Sunday, December 8th**

Useful References

1. CORAAL dataset:
   Kendall, Tyler and Charlie Farrington. 2023. *The Corpus of Regional African American Language.* Version 2023.06. Eugene, OR: The Online Resources for African American Language Project.
   https://oraal.github.io/coraal

2. Google Colab Notebook for feature extraction and classification:
   https://colab.research.google.com/drive/1hywRHp82IrepNMtWXYvcMmtX1MIibmQp

3. Package with all training and test files for the project:
   https://ucla.box.com/s/mohh4fnmgj3vekui8i8n28i02odleaop

4. Dialect Density Estimation in African American English:
   A. Johnson, K. Everson, V. Ravi, A. Gladney, M. Ostendorf, and A. Alwan, *Automatic Dialect Density Estimation for African American English*, in Interspeech 2022

5. The difficulties of working with accented audio:
   A. Koenecke, A Nam, E. Lake, J. Nudell, M. Quartey, Z. Mengesha, C. Toups, J. R. Rickford, D. Jurafsky, and S. Goel, *Racial disparities in automated speech recognition*. Proceedings of the National Academy of Sciences, Apr 2020

6. Techniques for accent classification:
   R. Huang, J. H. L. Hansen and P. Angkititrakul, *Dialect/Accent Classification Using Unrestricted Audio*, in IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 2, Feb. 2007

7. Methods for speaker identification:
   D. Snyder, D. Garcia-Romero, G. Sell, D. Povey and S. Khudanpur, *X-Vectors: Robust DNN Embeddings for Speaker Recognition*, 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

8. Self Supervised Representations for Accent Identification:
   K. Chang., Y.-H. Chou., J. Shi, H.-M. Chen, N. Holliday, O. Scharenborg, D.R. Mortensen (2024) *Self-supervised Speech Representations Still Struggle with African American Vernacular English*, Proc. Interspeech 2024, 4643-4647

9. Useful toolkit for speech processing and feature extraction:
   https://librosa.org/