11-785 PROJECT PROPOSAL: SPEAKER IDENTIFICATION USING DEEP NEURAL NETWORKS

Fan Hu
fanh@andrew.cmu.edu

Linwu Zhong
linwuz@cs.cmu.edu

Siyuan Wang
siyuanw@cs.cmu.edu

Zinian Zhao
zinianz@andrew.cmu.edu

ABSTRACT

The project aims to develop a model to identify if two speech recordings belong to the same speaker. Different feature extraction and representation methods as well as various neural network architectures will be compared and investigated to implement an optimal model, which hopefully can achieve an equal error rate of less than 1%.

1 INTRODUCTION

In this project, we would like to work on NIST dataset and build a deep neural network (DNN) model to identify if two speech recordings belong to the same speaker. The project scope includes finding optimal feature representations from the acoustic data, exploring the best network and architecture for the model, and also identifying the most suitable loss function etc.

2 BASELINE OR INITIAL EXPERIMENTS

As we have just obtained the dataset at the point of writing this proposal, we are not able to run preliminary experiments with data yet. However, we did a summary of the dataset. The training data is over 50GB obtained from SRE2008, and we are going to get more data of other years. Each file is an individual speech recording labelled with speaker ID. Enrollment dataset is 57GB, which consists of one speech recording for each speaker. Test data is much smaller with only 719 recordings. The setup for the experiments is that we randomly match the enrollment recordings with test recordings, which result in 416,119 trial pairs. The model is only trained on training data and then experiments will be conducted on the trail pairs.

We did some literature review on prior work on the task. One of the established baseline approach is the Gaussian Mixture Speaker Models introduced by Reynolds & Rose (1995). The individual Gaussian components of a GMM are shown to represent some general speaker-dependent spectral shapes that are effective for modeling speaker identity. According to Lei et al. (2014), GMM with universal background model (UBM) can achieve Equal Error Rate of 1.99 on NIST 2012 dataset, which should be similar to NIST 2008 dataset in terms of difficulty. Some other baseline models are also available, such as the total variability model and PLDA.

3 PROOF OF CONCEPT

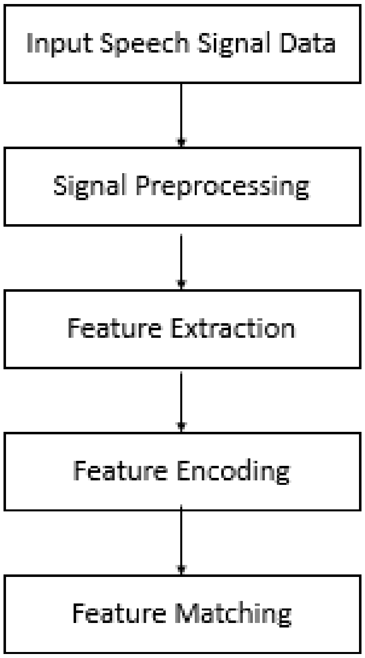

Figure 1: System Architecture Overview

Figure 1 shows the proposed architecture for our speaker identification system, which includes 4 main modules: (1) Signal Pre-processing (2) Feature extraction (3) Feature Encoding (4) Feature Matching

3.1 SIGNAL PRE-PROCESSING

As outlined by Keshvi Kansara (2016), signal pre-processing mainly includes (1) Silence removal (2) Pre-emphasis (3) Framing (4) Windowing.

A. SILENCE REMOVAL

In the original speech signal, there might be some silence part at the beginning and at the end, or in between two sentences. This silence data contains no useful information, thus could be removed to make the data cleaner.

B. PRE-EMPHASIS

In general, the high frequencies components in the original speech signal have smaller magnitude than the low frequencies components. In order to balance the frequency spectrum, a pre-emphasis filter could be applied on the speech signal to enhance the high frequencies components.

C. FRAMING

The frequencies in the speech signal are not always stationary over a long period of time. However, we could consider the signal to be stationary over a short period of time. In order to keep the frequency contours of the speech signal, we could divide the signal into short time frames with an overlap of a small frame size.

D. Windowing

After dividing the signal into short time frames, in order to smoothen each frame of the signal, we could apply some window function such as Hamming window to each frame.

3.2 Feature Extraction

Mel-frequency cepstral coefficients (MFCC) approach is used for feature extraction as suggested by Keshvi Kansara (2016). The whole process of feature extraction mainly includes (1) Fast Fourier Transform (FFT) (2) Filter Banks (3) Cepstrum Construction (4) Mean Normalization

A. Fast Fourier Transform (FFT)

The first step of feature extraction is to do Fast Fourier Transform (FFT), which converts the signal data of each frame from time domain to frequency domain. We can then calculate the power spectrum based on the FFT result.

B. Filter Banks

After applying FFT and calculating the power spectrum, we could apply filters on a Mel-scale to extract frequency bands. The reason to use Mel-scale is due to that human ear perception of the frequency contents for speech signals is not in a linear scale.

C. Mel-frequency Cepstral Coefficients (MFCC)

The filter bank coefficients calculated in Step B could be highly correlated. In order to decorrelate the filter bank coefficients, we could apply Discrete Cosine Transform (DCT) method to get a reduced number of cepstral coefficients.

D. Mean Normalization

In order to improve the Signal-to-Noise (SNR), we could also do mean normalization on MFCCs.

3.3 Feature Encoding

Since different speech recordings tend to vary in length, a fixed-length vector that could represent important information about the speaker and all other types of variability in a given speech segment would be very helpful. Dehak et al. (2011) proposed a model that could represent each recording using a low-dimensional vector named i-vector via factor analysis. Actually, i-vectors could be used as both the direct input of the final decision generation and the intermediate output of further encoding.

Besides the i-vector model, an autoencoder could be further deployed on the raw i-vectors so as to reduce the within-speaker variability Pekhovsky et al. (2016). In this project, we plan to represent each speech recording as a fixed-length i-vector in the first place. Then, an autoencoder will be used to learn a better representation that not only ensures that utterances spoken by the same speaker would have similar encodings, but also maximizes the scatteredness of all encodings.

3.4 Feature Matching

There are several available ways to achieve feature matching. The most naive method is simply computing every trial pair's cosine similarity. As the utterances given by the same speaker tend to have similar encodings, a cosine kernel could be used to generate a final decision score that indicates the probability of the two utterance recordings belong to the same speaker. If the decision score is above the threshold, the model will make the decision.

Besides cosine similarity, PLDA could be used to linearly discriminate between speakers in a low-rank subspace and measure the decision score in a probabilistic sense. Another option is using SVM (cosine kernel) or neural network to do binary classification based on the vector representations of the two recordings. However, this method is a little bit more complex than cosine similarity, as the enrollment data is always needed for the classification task here.

4 Final Experiments

All the experiments will be carried out on the whole NIST dataset collected, which consists of several years' NIST data. The model will be trained on the training data and tested on the trial pairs, which consist of the test data and enrollment data. The performance of the model will be mainly evaluated by equal error rate. In the experiment, each utterance will be represented by several overlapping frames. The frame span here is 25ms, while the frame shift is 10ms. With MFCC, each speech frame is represented by a 40-dimensional feature vector. In the end, utterances will be transformed and represented as fix-length i-vectors.

Since we have discussed several possible methods to do feature matching, all of the method combinations will be trained and tested on the dataset. The combination that achieves the best performance will be chosen as the final model setting. In order to prove the effectiveness of the proposed model, several baseline models will be evaluated on the same dataset, such as GMM-UBM, the total variability model and PLDA. The UBM model here will be a gender-independent model containing 2048 Gaussians. For the total variability model, we will only choose cosine similarity to be the scoring function as it outperformed SVM in the experiments discussed by Dehak et al. (2011). Through the comparison with these well-defined baseline models, the effectiveness of the proposed model will be fully validated.

5 Final Goals & Evaluation

The ultimate goal of this project is to develop a state-of-the-art model that is able to identify if the two speech recordings belong to the same speaker based on our training data and meanwhile the model has to achieve an equal error rate of approximately 1%.

To evaluate our experiments relative to these end goals, Equal Error Rate (EER) will be the single determiner to measure the final result of the model. Besides, we will employ the following quantitative measures as hard metrics during experimental process to compare performances between different systems:

- accuracy
- precision
- recall
- false alarm rate at a miss rate of 10% (FA@M10)

6 Related Work

6.1 Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models

This paper, published in 1995, introduces the use of Gaussian mixture models (GMM) for robust text-independent speaker identification. It is a strong baseline as well as one of the great traditional methods for our task.

6.2 Front-End Factor Analysis for Speaker Verification

This paper, published in 2011, introduces a i-vector model that could represent variable-length speech recordings via fixed-length vectors while the channel variabilities induced by different sources are compensated and the speaker characteristic preserved.

6.3 Bayesian Speaker Verification with Heavy-Tailed Priors

This talk, given in 2010, introduces a probabilistic latent discriminant analysis method that could linearly discriminate between speakers in a low-rank subspace given two raw i-vectors.

6.4 On Autoencoders in the i-vector Space for Speaker Recognition

This paper, published in 2016, introduces a method that utilizes autoencoder to learn a better encoding representation based on i-vectors.

6.5 Discriminative Autoencoders for Speaker Verification

This paper, published in 2017, introduces a loss function of an autoencoder which takes three kinds of error into consideration: reconstruction error, encoding similarity of utterances spoken by the same speaker and scatteredness of all encodings.

6.6 Speaker Recognition Using MFCC and Combination of Deep Neural Networks

This paper, published in 2016, introduces the process flow of signal pre-processing and feature extraction using MFCCs.

7 Data & Technical Requirements

We plan to store the dataset on Amazon S3. We have obtained a subset (SRE2008) of training data with the help of Professor Bhiksha Raj. We are going to obtain the full SRE dataset as the project progresses. As for software libraries, PyTorch will be used for neural network modeling. Scipy libraries like scipy.io.wavfile and scipy.fftpack import dct will be used for signal processing and feature extraction.

References

Najim Dehak, Patrick J. Kenny, Reda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19:788–798, 2011.

Patrick Kenny. Bayesian speaker verification with heavy-tailed priors. *Odyssey*, 2010.

Dr. A.C. Suthar Keshvi Kansara. Speaker recognition using mfcc and combination of deep neural networks. *International Journal of Advance Research and Innovative Ideas in Education*, 2: 3008–3013, 2016.

Hung-Shin Lee, Yu-Ding Lu, Chin-Cheng Hsu, Yu Tsao, Hsin-Min Wang, and Shyh-Kang leng. Discriminative autoencoders for speaker verification. *Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5375–5379, 2017.

Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren. A novel scheme for speaker recognition using a phonetically-aware deep neural network. *Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1695–1699, 2014.

Timur Pekhovsky, Sergey Novoselov, Aleksei Sholohov, and Oleg Kudashev. On autoencoders in the i-vector space for speaker recognition. *Proc. Odyssey*, pp. 217–224, 2016.

Douglas A. Reynolds and Richard C. Rose. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3:72–83, 1995.