Digital Signal Processing: Speaker Recognition Progress Report

Xinyu Zhou, Yuxin Wu, and Tiezheng Li
Tsinghua University

1 Introduction

In this week, we've done some fundamental work on the speaker recognition task, and gained some valuable result. Our present work consists of the following part:

1. Simple implementation of a small speaker recognition program, using MFCC and GMM.
2. Manage to get and preprocess the MOCHA-TIMIT corpus database.
3. Test the performance of our algorithm.

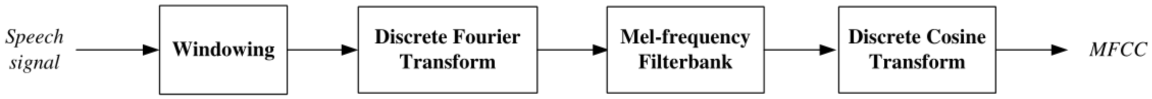
2 Algorithm and Implementation

We presented a prototype system based on MFCC as acoustic features, and GMM as our recognition model.

2.1 MFCC

MFCC (Mel-frequency Cepstral Coefficient) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel-scale of frequency [1] . MFCC is the mostly widely used features in Automatic Speech Recognition(ASR), and it can also be applied to Speaker Recognition task.

The process to extract MFCC feature is as followed:



First, the input speech should be divided into successive short-time frames of length L, neighboring frames shall have overlap R. In our implementation, We choose $L = 20ms$ ans $R = 10ms$. Those frames are then windowed by Hamming Window, as shown in Figure??

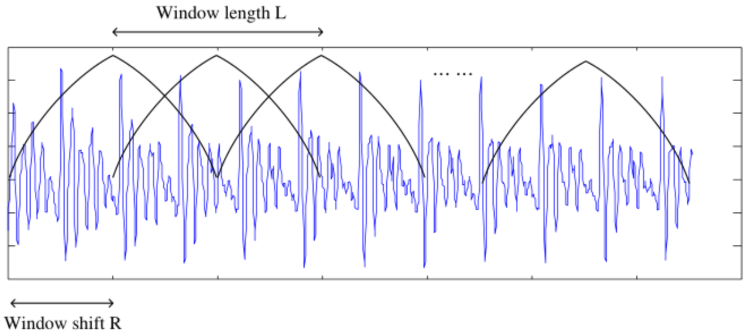
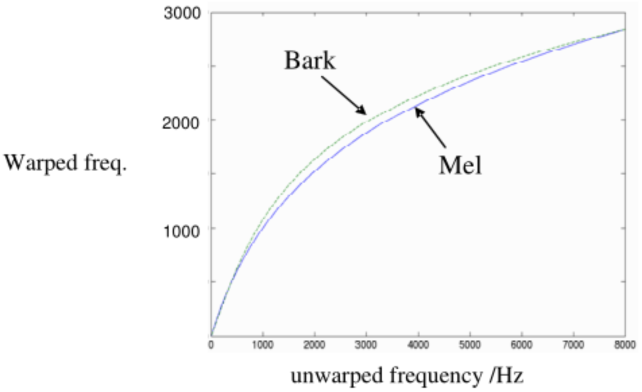


Figure 1: Framing and Windowing

Then, We perform Discrete Fourier Transform (DFT) on windowed signals to compute their spectrums. For each of N discrete frequency bands we get a complex number $X[k]$ representing magnitude and phase of that frequency component in the original signal.

Considering the fact that human hearing is not equally sensitive to all frequency bands, and especially, it has lower resolution at higher frequencies. Scaling methods like Mel-scale and Bark-scale are aimed at scaling the frequency domain to fit human auditory perception better. They are approximately linear below 1 kHz and logarithmic above 1 kHz.



In MFCC, Mel-scale is applied on the spectrums of the signals. The expression of Mel-scale warpping is as followed:

$$M(f) = 2595 \log_{10}(1 + \frac{f}{700})$$

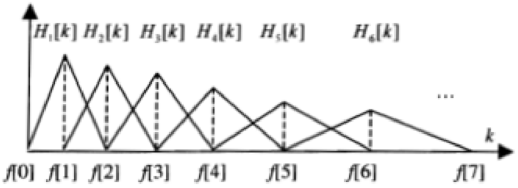

$H_1[k]$ $H_2[k]$ $H_3[k]$ $H_4[k]$ $H_5[k]$ $H_6[k]$
…
k
$f[0]$ $f[1]$ $f[2]$ $f[3]$ $f[4]$ $f[5]$ $f[6]$ $f[7]$

Figure 2: Filter Banks (6 filters)

Then, we appply the bank of filters according to Mel-scale on the spectrum, calculate the logarithm of energy under each bank by $E_i[m] = \log(\sum_{k=0}^{N-1} X_i[k]^2 H_m[k])$ and apply Discrete Cosine Transform (DCT) on $E_i[m](m = 1, 2, \cdots M)$ to get an array c_i:

$$c_i[n] = \sum_{m=0}^{M-1} E_i[m] \cos(\frac{\pi n}{M}(m - \frac{1}{2}))$$

Usually, the first 13 terms in c_i is used as features for future training.

2.2 GMM

We use Gaussian Mixture Model (GMM) to model all features from one person. For implementation, we use the GMM model training and predicting routine from the famous python machine learning package scikit-learn [2]. Since the last step of MFCC is DCT, different dimensions of the features are strongly independent, so we use GMM with diagonal covariance matrix. The number of components in GMM is chosen as 32 in our implementation.

After building models for each person, it can be used to calculate the probability that the input signal is generated by this model. The model with maximum probability is picked out as the result, and the corresponding person is recognized.

res/test.py

```python
def cal_score(model, mfcc):
    return np.exp(sum(model.score(mfcc)) / 1000)

def pred_label(mfcc):
    scores = [cal_score(gmm, mfcc) for gmm in gmms]
    return max(enumerate(scores), key=operator.itemgetter(1))[0]
```

3 Test

The corpus we use comprise of three different speakers: two men and one woman. Each speaker has 460 samples, where the average duration of each sample is about 4 seconds. For each speaker, we randomly choose M samples to train GMM model. For more deep inspection of the algorithm, M is enumerated from 2 to 5. We then feed the whole corpus to to the algorithm, and test its accuracy. Further more, for the robustness of the result, we repeat the test process 10 times, and average the result.

3.1 Result

#Training sample	2	3	4	5
Accuracy	0.968555	0.991293	0.997549	0.997380

As we can see from the curve ploted, the performance of the algorithm increases as the number of training samples given increases. The accuracy when just using two training samples for each user has reached 96.55%, which is supprisingly high with respect to using such small amount of training samples.

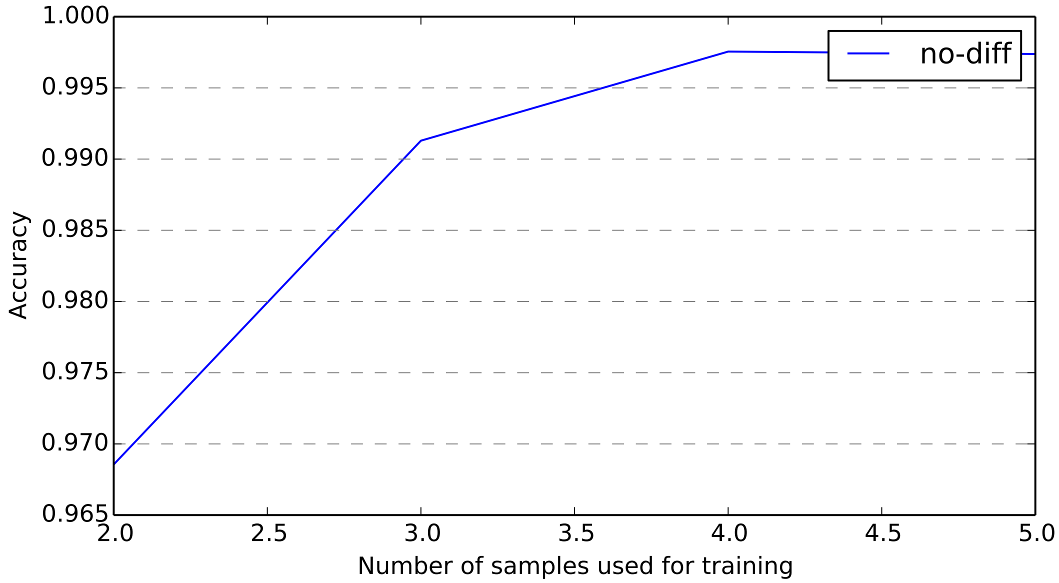


Figure 3: Accuracy vs Number of samples used for training

When using 4 to 5 samples per speaker, the accuracy is above 99.73%, which strongly confirmed the effectivenes of MFCC feature and GMM modeling for each speaker.

However, further inspection on misclassified samples showed limitation on our test case: all the misclassified samples are within two men speakers, indicates the weakness of the algorithm.

4 Future Work

A text-independent speaker-recognition system is presented at this period. But our current still have lots of limitations. Due to our time devoted in, we only manage to acquire a small corpus on the Internet with only 3 people. The data in the corpus is quite clear, thus we may need to do some further test on the robustness of our algorithm.

In the future, we are focusing on variables in input limitation. The recognizability of speaker can be affected not only by the linguistic message but also the age, health, emotional state and effort level of the speaker. Background noise and performance of recording device also interfere the classification process.

A goal we want to achieve in this project can be described as follows: For a clear conversation signal between two persons, with no sentences overlapped and significant interval between sentences, we may separate the conversation signal into two parts. Each part is exactly all the sentences spoken by a certain speaker.

A main challenge is that MFCC's performance decreases incredibly when the strength of background noise enhances. A series of speech noise reduction algorithms needs to be applied on the input signals. Fortunately, much work in this field has been done previously. Spectral Subtraction(SS) that incorporates noise over time, and SPLICE algorithm that makes no assumptions about noise stationarity can be applied to build a noise-robust recognition system as two examples. Another challenge settles in separating overlapped sentences from two or more different speakers. Vocal separation is a challenging problem, to date there is no general algorithm that ensures perfect separation effect. Some commonly used models and algorithms are listed as follows: NSA, LVD/PLCA, SFD, etc. More work on this field will be done till next period.

References

[1] *MFCC - Wikipedia, the free encyclopedia*. URL: http://en.wikipedia.org/wiki/Mel-frequency_cepstrum.

[2] *scikit-learn – Machine Learning in Python*. URL: http://scikit-learn.org/stable/.