PROJECT REPORT
COURSE: SPEECH PROCESSING (INT3411 20)

SPEAKER IDENTIFICATION USING GAUSSIAN MIXTURE MODELS (GMM)

Nguyen Huy Hoang
Nguyen Xuan Hoang

June 2021

CONTENTS

1 INTRODUCTION

In this project, we focus on a statistical method to answer the question 'who is the speaker' in the speech file. Recently, a lot of voice biometric systems have been developed which can extract speaker information from the recorded utterance and identify the speaker from set of trained speakers in the database. In this project, we will illustrate the same with a naive approach using Gaussian Mixture Models (GMM).

2 DATASET

We utilize the available dataset in this course and randomly choose 20 out of 70 speakers for the purpose of our project. In the original dataset, the length of each recording is about 3-5 minutes, which is computationally expensive if we use the whole audio file to extract MFCC and delta features. Therefore, for each recording, we split it into smaller set of 5-6 seconds in length for the sake of feature extraction.

Training data: It consists of 5 audio files for each speaker, spoken by 20 speakers. The length of each audio file is about 5-6 seconds/speaker.

Testing data: This consists of remaining 5 **unseen** audio files of the same 20 speakers. All audio files are also of 5-6 seconds duration.

In total, the dataset includes 200 audios of 20 speakers and is equally divided for training and testing.

3 FEATURE ENGINNERING

3.1 Feature Extraction

To create an acoustic model, our observation X is represented by a sequence of acoustic feature vectors $(x_1, x_2, x_3, \dots)$
We also hope the extracted features will be robust to who the speaker is, and the noise in the environments. Also, like any ML problems, we want extracted features to be independent of others. It is easier to develop models and to train these models with independent features.
One popular audio feature extraction method is the Mel-frequency cepstral coefficients (MFCC). The feature count is small enough to force us to learn the information of the audio. It provides us enough frequency channels to analyze the audio.
Below is the flow of extracting the MFCC features.

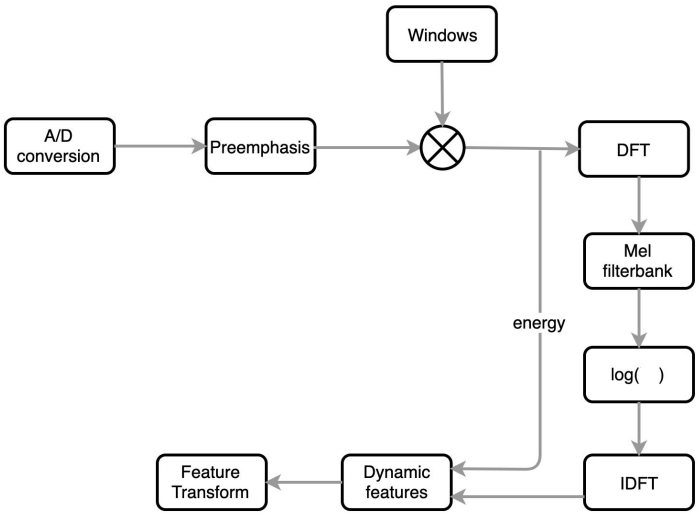

Figure 1: MFCC Pipeline

- **Cepstrum — IDFT**: Cepstrum is the reverse of the first 4 letters in the word "spectrum". For speech recognition, we just need the coefficients on the far left and discard the others. In fact, we take the first 20 cepstral values. Mathematically, the transformation produces uncorrelated features. Therefore, MFCC features are highly unrelated. In ML, this makes our model easier to model and to train. If we model these parameters with multivariate Gaussian distribution, all the non-diagonal values in the covariance matrix will be zero. Mathematically, the output of this stage is

$$c[n] = \sum_{n=0}^{N-1} \log\left(\left|\sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}\right|\right) e^{j\frac{2\pi}{N}kn} \tag{1}$$

- **Dynamic features (delta)**: We extract 40-dimensional features from speech frames. There are 20 MFCC features and 20 derivatives of MFCC features. The derivatives of MFCCs provides the information of dynamics of MFCCs over the time. It turns out that calculating the delta-MFCC and appending them to the original MFCC features (20-dimentions) increases the performance in lot of speech analytics applications. To calculate delta features from MFCCs, we apply the following equation.

$$d_t = \frac{\sum_{n=1}^{N} n\left(c_{t+n} - c_{t-n}\right)}{2\sum_{n=1}^{N} n^2} \tag{2}$$

where 'N' is number of deltas summed over. Typically taken as 2.

3.2 MFCC Normalization

Mel Frequency Cepstral Coefficients (MFCC) are widely used in speech recognition and speaker identification. MFCC features are usually pre-processed before being used for recognition. One of these pre-processing is creating delta and delta-delta coefficients and append them to MFCC to create feature vector. Another pre-processing is coefficients mean normalization, it reduces varying channel and noise effects occuring in the audio data
In our project, we use both of 2 techniques mentioned above. By experiment, it shows that by applying mean normalization, our accuracy score increases greatly on the test set and it does not got bad result when the data comes from different sources. For example, the predicted result is not affected by recording by mobile phone in the train set and recording by laptop or other devices in the test set . Therefore, we can conclude that MFCC Normalization is a must-have step to get a high result on Speaker identification task.

4 GAUSIAN MIXTURE MODELS (GMM)

4.1 What is GMM?

A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of Gaussian distributions.
This allows the distributions to be multimodal, i.e. we allow a feature to have a few possible values. This provides the flexibility of variants in speech.
A GMM will take as input the MFCCs and derivatives of MFCCs of the training samples of a speaker and will try to learn their distribution, which will be representative of that speaker
For K=2, we will have 2 Gaussian distributions $G_1=\left(\mu_1, \sigma_1^2\right)$ and $G_1=\left(\mu_1, \sigma_1^2\right)$. In our task, we initilize K=16. We start with random initialization of parameters μ and σ and compute the probability on which cluster that a data point may belong to. Then, we recompute the parameters and for each Gaussian distribution based on this probability. The data points are re-fitted to different clusters and the Gaussian parameters are re-calculated again. The iterations continue until the solution converges.

4.2 E-M Algorithm

The EM algorithm alternates between performing an expectation estimation (E-step) and a maximization (M-step).

- Initialize the $G1$ and $G2$'s parameters $\left(\mu_1, \sigma_1^2\right)$ and $\left(\mu_2, \sigma_2^2\right)$ with random values. Set $P(a) = P(b) = 0.5$

- For all the training datapoints $x_1, x_2, \cdots$, compute the probability that it belongs to $a\left(G_1\right)$ or $b\left(G_2\right)$.

$$P\left(x_i \mid b\right) = \frac{1}{\sigma_b\sqrt{2\pi}} e^{-(x-\mu_b)^2/2\sigma_b^2}$$

$$b_i = P\left(b \mid x_i\right) = \frac{P\left(x_i \mid b\right) P(b)}{P\left(x_i \mid b\right) P(b) + P\left(x_i \mid a\right) P(a)}$$

$$a_i = P\left(a \mid x_i\right) = 1 - b_i$$

- Now, we recalculate the parameters for G_1 and G_2

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \cdots + b_n x_n}{b_1 + b_n + \cdots + b_n}$$
$$\sigma_b^2 = \frac{b_1 (x_1 - \mu_b)^2 + \cdots + b_1 (x_n - \mu_b)^2}{b_1 + b_2 + \cdots + b_n}$$
$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \cdots + a_n x_n}{b_1 + a_n + \cdots + a_n}$$
$$\sigma_a^2 = \frac{a_1 (x_1 - \mu_a)^2 + \cdots + a_1 (x_n - \mu_a)^2}{a_1 + a_2 + \cdots + a_n}$$

- Recalculate the priors

$$P(b) = \frac{b_1 + b_2 + \cdots + b_n}{n}$$
$$P(a) = 1 - P(b)$$

- For Gaussian Distribution with multiple variate, the probability distribution function is:

$$P\left(x_i \mid b\right) = \frac{1}{\sqrt{(2\pi)^n \left|\Sigma_b\right|}} e^{-\frac{1}{2}(x-\mu_b)^T \Sigma_b^{-1} (x-\mu_b)}$$

4.3 GMM Parameters

The parameters used in our code can be described in table 1 below.

Parameters	Value	Function
n_components	16	Number of Gaussian models
n_iter	200	Number of iterations to be performed for estimating the parameters of these n components
covariance_type	diag	Type of co-variance to be assumed between the features
n_init	3	Number of initializations to perform. The best results are kept.

Table 1: Parameters used in GMM

5 EVALUATION METHOD

While testing when the speaker of a new voice sample is to be identified, first the 40-dimensional feature (MFCCs + delta MFCC) of the sample will be extracted and then the trained speaker GMM models will be used to calculate the scores of the features for all the models. Speaker model with the maximum score is predicted as the identified speaker of the test speech.

Test set consists of 5 unseen utterances of trained 20 speakers

Upon arrival of a test voice sample for speaker identification, we begin by extracting the 40 dimensional for it, with 25 ms frame size and 10 ms overlap between frames. Next we require the log likelihood scores for each frame of the sample, x_1, x_1, ... ,x_i , belonging to each speaker, ie, $P(x_i|S_j)$ (for all j that belongs to S) is to be calculated. The likelihood of the frame being from a particular speaker is calculated by substituting the μ_1 and Σ of that speaker GMM model in likelihood equation shown in previous section. This is done for each of the 'k' Gaussian components in the model, and the weighted sum of the 'k' likelihoods from the components is taken as per the weight 'w ' parameter of the model. The logarithm operation when applied on the obtained sum gives us the log likelihood value for the frame. This is repeated for all the frames of the sample and the likelihoods of all the frames are added. The speaker model with highest likelihood score is considered as the identified speaker.

6 RESULTS AND CONCLUSION

We achieve an accuracy of 98%, identifying 98 out of 100 speech utterances correctly. There are few reasons for such perfect result.

1. The unseen utterances of speakers taken from the dataset are possibly of same channel or environment.
2. The evaluation task in performed on small dataset. Consider the situation when we have to identify speakers from the set of 1000 speakers.
3. In this evaluation, we have not taken out-of-set speakers into account i.e. if the audio is not from any speaker still our system will identify it as one of speakers in trained set depending upon highest likelihood.
4. In the real environment, we may get more noisy and unclean data. Speaker identification system needs to be robust.

7 CONTRIBUTION

Student	ID	Contribution
Nguyen Huy Hoang	18020557	50%
Nguyen Xuan Hoang	18020544	50%