Comparative Study of Speaker Recognition Methods: DTW, GMM and SVM

Loh Mun Yee Abdul Manan Ahmad
Faculty of Computer Science & Information System
University of Technology Malaysia
Skudai, 81310 Johor Darul Ta’zim, Malaysia

Contact No: +60123572086
Email: lohmunyee@gmail.com

Contact No: + 07-5532201
Email: manan@utm.my

ABSTRACT
Speaker recognition is a process where a person is recognized on the basis of his/her voice signals. The problem of speaker recognition belongs to a much broader topic in scientific and engineering so called pattern classification. In this paper we provide a brief overview for evolution of pattern classification technique used in speaker recognition. We also discussed about our propose process to modeling a speaker recognition system, which include pre-processing phase, feature extraction phase and pattern classification phase. Besides, we aim at concerns a comparison of DTW, GMM and SVM for speaker recognition and experimental result for these 3 techniques are presented in this paper. Experiments in this study were performed using TIMIT speech database. Experimental result shows that SVM gain the worse result among 3 types of classifier. This is due to the drawback of SVM when dealing with audio data is their restriction to work with fixed-length vectors. Hence, investigation on a better normalization function will be done to ensure that the SVM classifier get the better accuracy rate.

Keywords
Speaker Recognition System, Dynamic Time Warping (DTW), Gaussian Mixture Model (GMM), Support Vector Machine (SVM).

1. INTRODUCTION
Speaker recognition is a process where a person is recognized on the basis of his/her voice signals [1]. Speaker recognition can be further broken into two categories: speaker identification and speaker verification. Identification takes the speech signal from an unknown speaker and compares this with a set of valid users. The best match is then used to identify the unknown speaker. Similarly, in verification the unknown speaker first claims identity, and the claimed model is then used for identification. If the match is above a predefined threshold, the identity is accepted.

The problem of speaker recognition belongs to a much broader topic in scientific and engineering so called pattern classification. The goal of pattern classification is to classify objects of interest into a number of categories or classes. The objects of interest are generically called patterns and in our case are sequences of acoustic vectors that are extracted from an input speech. The classes here refer to individual speakers [2]. Pattern classification plays as a crucial part in speaker modeling component chain. The result of pattern classification will strongly affect the speaker recognition engine to decide whether to accept or reject a speaker.

Many research efforts have been done in speaker recognition pattern classification. There are Dynamic Time Warping (DTW) [3], Vector Quantization (VQ) [4], Hidden Markov Models (HMM) [5], Gaussian mixture model (GMM) [6] and so forth. Most of this works are based on generative model. A generative model is a model for randomly generating observed data, typically given some hidden parameters. Because of the randomly generating observed data functions, they are not able to provide a machine that can directly optimize discrimination.

In fact, pattern classification should manage to optimize discrimination. Consequently, Support Vector Machine (SVM) was introducing as an alternative classifier for speaker verification [7]. SVM, which are based on the principle of structural risk minimization, consist of binary classifiers that maximize the margin between two classes. The power of SVM lies in their ability to transform data to a higher dimensional space and to construct a linear binary classifier in this space. It sounds efficient and useful to speaker recognition application, but they cannot easily deal with the dynamic time structure of sounds, since they are constrained to work with fixed-length vectors [8]. When working with audio signals, each signal frame is converted into a feature vector of a given size, the whole acoustic event is represented by a sequence of feature vectors, which shows variable length.

In the work reported in this paper, we aim at concerns a comparison of DTW, GMM and SVM for speaker recognition. The emphasis of the experiments is on the performance of the models under incremental amounts of training data in an attempt to identify the best approach for hybrid SVM in order to improve SVM problem as just stated as paragraph above.
This paper is organized as follows. In Section 2, we quickly review our propose speaker recognition structure. In Section 3, we discuss the methods we use for preprocessing signal and section 4 shows the feature extraction techniques. Section 5 discusses how we construct DTW, GMM and SVM for speaker recognition. Section 6 shows the experimental result for these 3 techniques. Finally, section 7 concludes our work.

2. OUR SPEAKER RECOGNITION FRAMEWORK

In this section, we generally review our propose speaker recognition structure. Our Speaker recognition system involves two main stages, the enrolment stage and the verification stage. These phases involve three main parts:

- Pre-Processing.
- Feature Extraction.
- Pattern Classification.

A block diagram of this procedure is shown in Figure 1. At the time of enrollment, speech sample is acquired in a controlled and supervised manner from the user. The speaker recognition system has to process the speech signal in order to extract speaker discriminatory information from it. This discriminatory information will form the speaker model. At the time of verification a speech sample is acquired from the user. The recognition system has to extract the features from this sample and compare it against the models already stored beforehand. This is a pattern matching or classification task.

Feature extraction maps each interval of speech to a multidimensional feature space. This sequence of feature vectors $X_i$ is then compared to speaker models by pattern classification. This results in a match score $Z_i$ for each vector or sequence of vectors. The match score measures the similarity of the computed input feature vectors to models of the claimed speaker or feature vector patterns for the claimed speaker. Last, a decision is made to either accept or reject the claimant according to the match score or sequence of match scores, which is a hypothesis testing problem.

3.0 PRE-PROCESSING

All speech data will perform in a discrete-time speech signal because of recorded by sampling the input. Therefore, we need some pre-processing techniques to make the discrete-time speech signal more flexible for the processes that follow. There are 4 pre-processing techniques that we before feature extraction. These include DC offset removal, silence removal, pre-emphasis and windowing.

3.1 DC Offset Removal

Speech data are discrete-time speech signal, it often carry some redundant constant offset called DC offset [9]. These DC offset will effect quality of the information extracted from the speech signal. Consequently, we calculating the average value of the speech signal and subtracting this from itself.

3.2 Silence Removal

This process is performed to discard silence periods from the speech containing silence frames. The signal energy in each speech frame is evaluated by equation (1).

$$E_i = \sqrt{\sum_{k=1}^{M} x(k)^2} \quad i = 1, \ldots, N$$  (1)

Where $M$ is the number of samples in a speech frame and $N$ is the total number of speech frames. Threshold is successively performed to detect silence frames with a threshold level determined by equation (2).

$$\text{Threshold} = E_{\text{min}} + 0.1 (E_{\text{max}} - E_{\text{min}})$$  (2)

$E_{\text{max}} - E_{\text{min}}$ are the lowest and greatest values of the $N$ segment respectively.

3.3 Pre-emphasizing

Pre-emphasis is a technique used in speech processing to enhance high frequencies of the signal. The main purpose of pre-emphasizing is to spectrally flatten the speech signal that is to increase the relative energy of its high-frequency spectrum. There are two important factors driving the need for pre-emphasis. Firstly, the speech signal generally contains more speaker specific information in the higher frequencies [10]. Secondly, as the
speech signal energy decreases the frequency increases. This allows the feature extraction process to focus on all aspects of the speech signal. Pre-emphasis is implemented as a first-order Finite Impulse Response (FIR) filter defined as:

$$H(Z) = 1 - 0.95 Z^{-1} \quad (3)$$

Figure 4 shows the speech signal before pre-emphasizing process and Figure 5 shows the speech signal after pre-emphasizing process.

![Figure 4. Speech Signal before Pre-emphasizing](image)

![Figure 5. Speech Signal after Pre-emphasizing](image)

### 3.4 Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. A windowing function is used on each frame to smooth the signal and make it more amendable for spectral analysis. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. If we define the window as, where $N$ is the number of samples in each frame, then the result of windowing is the signal.

$$y_1(n) = x(n)w(n), \quad 0 \leq n \leq N-1 \quad (4)$$

Typically the Hamming Window is used, which is of the form

$$w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{N-1} \right), \quad 0 \leq n \leq N-1 \quad (5)$$

### 4.0 FEATURE EXTRACTION

After Davis and Mermelstein reported that Mel-frequency cepstral coefficients (MFCC) provided better performance than other features in 1980 [11], MFCC has been widely used as the feature parameter for automatic speaker recognition. In our implementation, we will use MFCC technique to extract the speech feature in order to obtain the best result for pattern classification. Figure 6 shows an outline of the process of MFCC.

![Figure 6. Outline of the process of MFCC](image)

As a result of the final step, 13 coefficients named MFCC for each frame are obtained. The 0th coefficient is not used because it represents the average energy in the signal frame and contains little or no usable information.

As the output of feature extraction phase, vectors in 12 dimensions are obtained for each frame. These vectors are used in pattern matching/classification technique for compare and match the feature sets against the model already stored before hand.

### 5.0 PATTERN CLASSIFICATION

The pattern classification task of speaker recognition involves computing a match score, which is a measure of the similarity of the input feature vectors to some model. Speaker models are constructed from the features extracted from the speech signal. To enroll users into the system, a model of the voice, based on the extracted features, is generated and stored (possibly on an encrypted smart card). Then, to authenticate a user, the matching algorithm compares/scores the incoming speech signal with the model of the claimed user.

In our experiment, three major speaker recognition pattern classification techniques has been chosen and a comparison of the performance has made. These techniques are DTW, GMM and SVM.

#### 5.1 Dynamic Time Warping

Dynamic time warping is an algorithm for measuring similarity between two sequences which may vary in time or speed. In our experiment, we use the DTW techniques which propose by Sadaoki Furui at years 1981[12]. According to Furui theory, the training data are used as a initial template, and the testing data is time aligned by DTW. DTW is a method that allows a computer to find an optimal match between two given sequences. The average of the two patterns is then taken to produce a new template to which a third utterance is time aligned. This process is repeated until all the training utterances have been combined into a single template.

The idea of the DTW technique is to match a test input represented by a multi-dimensional feature vector $T= [t_1, t_2...t_I]$ with a reference template $R= [r_1, r_2...r_J]$. While aim of DTW is to find the function $w(i)$, as shown in figure 7.
5.2 Gaussian Mixture Models

The Gaussian mixture model (GMM) is a density estimator and is one of the most commonly used types of classifier. In this method, the distribution of the feature vector $x$ is modeled clearly using a mixture of $M$ Gaussians.

$$P(x|M) = \sum_{i=1}^{M} a_i \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^\frac{1}{2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i)\right)$$

(6)

Here $\mu_i$, $\Sigma_i$ represent the mean and covariance of the $i^{th}$ mixture. Given the training data $x_1, x_2, ..., x_n$, and the number of mixture $M$, the parameters $\mu_i, \Sigma_i, a_i$ is learn using expectation maximization. During recognition, the input speech is again used extract a sequence of features $x_1, x_2, ..., x_L$. The distance of the given sequence from the model is obtained by computing the log likelihood of given sequence given the data. The model that provides the highest likelihood score will verify as the identity of the speaker. A detailed discussion on applying GMM to speaker modeling can be found in [6].

5.3 Support Vector Machines

SVM is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization. The decision surface is a weighted combination of elements of a training set. These elements are called support vectors, which characterize the boundary between the two classes. Let the two classes of the binary problem be labeled +1 and -1.

For the purpose to characterize the boundary between the two classes, we need maximizing the margin. Maximizing the margin are the process find the “middle-line” consider two parallel lines both of which separate the two classes without error. Several steps need to be determine the linear separator (Figure 8a, 8b, 8c):

- Find closest points in convex hulls
- Plane bisect closest points
- Maximize distance between two parallel supporting planes

Figure 8. Steps for Binary Linear Decision Boundary

During speaker recognition process, classifying the feature which derived from the transformation of feature extraction directly will not immediately works when using SVM [13]. It is because SVM only can process fixed-length input, whereas speech signals are non-stationary. Therefore, we need to categorizes the feature and scaling them.

SVM requires that each data instance is represented as a vector of real numbers. Hence, if there are categorical attributes, we first have to convert them into numeric data. We recommend using $m$ numbers to represent an $m$-category attribute. Only one of the $m$ numbers is one, and others are zero. For example, a two-category attribute such as {speaker, imposter} can be represented as (0,1) and (1,0).

Scaling them before applying SVM is very important. The main advantage is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large attribute values might cause numerical problems. We recommend linearly scaling each attribute to the range $[-1, +1]$ or $[0, 1]$.

6.0 EXPERIMENTAL SETUP & RESULTS

In this section, we describe the experiments carried out in order to test the different recognizers as stated as above. The emphasis of the experiments is on the performance of the models under incremental amounts of training data in an attempt to identify the best approach for hybrid SVM. Experiments are conducted on a clean condition. In orders to get a fair comparison between 3 types of classifier, for each of then we have properly selected the same datasets and done some pro-processing for enhanced the feature data through a set of preliminary experiments.

6.1 Dataset Description

We performed our evaluation on the TIMIT speech database. The TIMIT corpus of read speech has been designed to provide speech data for development and evaluation of automatic speech recognition systems. However, the large number of distinct speakers present in the system also makes it suitable for
evaluation speaker recognition system as well. TIMIT contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect region of United Stated. Out of this large set, we chose 5 utterances of 10 distinct users to evaluate our system.

6.2 Computer Tools used

The program was implemented in matlab 7.0 using the signal processing tool box.

6.3 DTW System Evaluation

The first method evaluated uses DTW as pattern classification techniques. To evaluate the system, each sample utterance of the user was compared with the rest of the utterances in the database. For each comparison, the distance measure was calculated. A lower distance measure indicates a higher similarity. It is also of interest to see the effect increasing the number of speakers on the accuracy results besides comparison classifier performance. The first set of experiments; we use the TIMIT corpus only for these results increasing the number of speakers from 10 to 50. Figure 9 shows the effect of increasing the speakers on performance of the DTW speaker verification system. Accuracy starts off highly 92% as would be expected, and slowly declines to approximately 80%. These results serve to show with increasing amounts of training data, the DTW distance measure become hard to calculated due to the progressively information of speaker.

6.4 GMM System Evaluation

The second method evaluated uses GMM as pattern classification techniques. Given training speech from a speaker’s voice, the goal of this speaker model training is to estimate the parameter og GMM, which in some sense best matches the distribution of the training feature vector. We use maximum likelihood (ML) estimation in our experiment. The aim of ML is to find the model parameters, which maximize the likelihood of the GMM given the training data. Therefore, the testing data which gain a maximum score will recognize as speaker.

The second set of experiments; we use the TIMIT corpus only for these results increasing the number of speakers from 10 to 50. Figure 10 shows the effect of increasing the speakers on performance of the GMM speaker verification system. Accuracy starts off highly 98% as would be expected, and slowly declines to approximately 83%, which is congruent with accuracy results found by Reynolds [6]. As can be observed, even GMM speaker verification accuracy rate has decrease when the training data increase, but it still obtain the better result if compare with DTW.

6.5 SVM System Evaluation

The third method evaluated uses SVM as pattern classification techniques. An SVM is essentially a binary classifier trained to estimated whether an input vector \( x \) belongs to a class 1 (the desired output would be then \( y=+1 \)) or to a class 2 (\( y=-1 \)) where class 1 is verify as speaker and class 2 is verify as imposter.

The third set of experiments; we use the TIMIT corpus only for these results increasing the number of speakers from 10 to 50. Figure 11 shows the effect of increasing the speakers on performance of the SVM speaker verification system. Accuracy starts off highly 72% as would be expected, and slowly declines to approximately 60%. As can be observed, SVM gain the worse result if compare with DTW and GMM.

6.6 Results

The result of our experiments shows in figure 12. The percentages rates of the accuracy are taken from the average data for 5 experiments run times. Results from experiments shows GMM likelihood function is a well understood statistical model whereas DTW suitable due with small fixed vocabulary system. As can be observed, SVM gain the worse result among 3 types of classifier. This is due to the drawback of SVM when dealing with audio data is their restriction to work with fixed-length vectors.
Obviously, the functions we choose for fixed-length vectors affect the performance of the SVM directly.

![Comparisons of 3 types of Speaker Verification Performance](image)

**Figure 11. Comparisons of DTW, GMM and SVM Performance**

7.0 CONCLUSION AND FUTURE RESEARCH WORK

In this paper, we have presented an introduction to the problem of speaker recognition and discussed our process to modeling a speaker recognition system. Besides, we explored three major pattern classification techniques for speaker recognition and described how it implements in speaker recognition system. In addition, we also evaluated their performance of the models under incremental amounts of training data.

From the experiments, we observe that one good way of applying SVM is to represent each utterance as a single N-dimensional vector, where N is fixed. In order to apply a SVM to this kind of data, one needs either to somehow normalize the size of the sequence of input space feature vectors or to find a suitable kernel function that can deal with sequential data.

Future work will be concentrating on investigation of the effectiveness of hybrid SVM for more robust speaker recognition. Investigation on a better normalization function also will be done to ensure that the SVM classifier get the better accuracy rate.

8.0 REFERENCES


