International Journal of Electronics and Electical Engineering

Volume 3 | Issue 3 Article 14

January 2015

APPLICATION OF GAUSSIAN SUPERVECTOR IN SPEECH ANALYSIS

KAULESHWAR PRASAD
BIT Durg, prasadkauleshwar@gmail.com

PIYUSH LOTIA
SSCET Bhilai, lotiapiyush@gmail.com

APPLICATION OF GAUSSIAN SUPERVECTOR IN SPEECH ANALYSIS

KAULESHWAR PRASAD[1], PIYUSH LOTIA[2]

[1]Asst. Professor, BIT Durg,
[2]Sr. Associate Professor, SSCET Bhilai

Abstract- The idea of the Speaker Identification is to implement a recognizer using Matlab which can identify a person by processing his/her voice. The basic goal of the paper is to classify and recognize the speeches of different persons. This classification is mainly based on extracting several key features like Mel Frequency Cepstral Coefficients (MFCC) from the speech signals of those persons by using the process of feature extraction using MATLAB. The above features may consists of pitch, amplitude, frequency etc. Using a statistical model like Gaussian mixture model (GMM) and features extracted from those speech signals we build a unique identity for each person who enrolled for speaker recognition. There is an elegant and powerful method for finding the maximum likelihood and that method is called Expectation and Maximization algorithm. The performance of the technique has been measured by three parameters: Number of Speakers in Database, Number of Persons Tested and the % Error.

Keywords- *Feature extraction, Mel Frequency Cepstral Coefficients, supervector, Gaussian Mixture Model (GMM)*

1. INTRODUCTION

Various researches have been performed in biometric recognition systems. Among the most popular traits are finger print, face, and voice. Each has pros and cons relative to accuracy and deployment. In voice recognition, there are two main factors that have made it a compelling biometric. First, speech is a natural signal to produce that is not considered threatening by users to provide. Second, the telephone system provides a ubiquitous, familiar network of sensors for obtaining and delivering the speech signal .During the last decade, speaker recognition technology has made its debut in several commercial products. Most deployed applications are based on scenarios with cooperative users speaking fixed digit string passwords or repeating prompted phrases from a small vocabulary.

These generally employ what is known as text-dependent or text-constrained systems. Such constraints are quite reasonable and can greatly improve the accuracy of a system; however, there are cases when such constraints become cumbersome or impossible to enforce.

An example of this is background verification where a speaker is verified behind the scene as he/she conducts some other speech interactions. For cases like this, a more flexible recognition system able to operate without explicit user cooperation and independent of the spoken utterance (called text-independent mode) is needed. Automatic speech recognition is therefore an engineering compromise between the ideal, i.e. a complete model of the human, and the practical, i.e. the tools that science and technology provide and that costs allow. At the highest level, all speaker recognition systems contain two main modules:

1.1 Feature Extraction :

Feature extraction is the process that extracts a small amount of data from the voice signal that can later be used to represent each speaker. It is also called front end analysis. There are three major types of front-end processing techniques, namely linear predictive coding (LPC), Mel-frequency Cepstral coefficients (MFCC)[4,5], and perceptual linear prediction (PLP), where the latter two are most commonly used in state-of-the-art ASR systems. Generally MFCC is used for feature extraction.

1.2 Feature Matching :

Feature matching involves the actual procedure to identify the unknown speaker by comparing extracted features from his/her voice input with the ones from a set of known speakers. The state-of-the-art in feature matching techniques used in speaker recognition includes Dynamic Time Warping (DTW), Hidden Markov Modeling (HMM), and Vector Quantization (VQ)[4]. The technique of VQ consists of extracting a small number of representative feature vectors as an efficient means of characterizing the speaker specific features .By means of VQ, storing every single vector that we generate from the training is impossible. By using these training data features are clustered to form a codebook for each speaker. In the recognition stage, the data from the tested speaker is compared to the codebook of each speaker and measure the difference.

These differences are then use to make the recognition decision. In vector Quantization each utterances could be represented as a single vector .The average vectors would then be compared using a distance measure which is computationally very efficient but gives poor recognition accuracy. A robust way to present utterances using a single vector called supervector [2 ,3 , 6].

2. GAUSSIAN SUPERVECTOR

The UBM is trained using the background databases that are selected to reflect the alternative imposter speeches. The EM algorithm is used for the UBM training. The GMM probability density can be described as follows:

$$p(x) = \sum_{i=1}^{M} w_i \, f(x|m_i,\Sigma_i) \quad (1)$$

Where x is a -dimensional cepstral feature vector, and m_i, $\sum_I$, w_i, (I = 1 …M) are, respectively, the mean vector, the co-variance matrix, and the weight of the Gaussian component. F (.) denotes Gaussian density function, i.e.,

$$`(x \mid m_i, \Sigma_i) = \frac{(2\Pi)^{-D/2}}{|\Sigma_i|^{1/2}} \times \exp\left(-1/2(x-m_i)^T \Sigma_i^{-1}(x-m_i)\right) \quad (2)$$

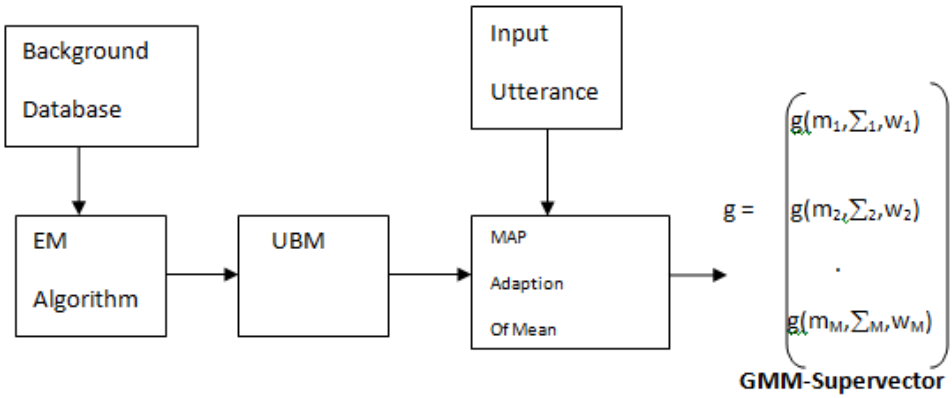

Figure 1. Process of generating the GMM supervector from an utterance.

g (m_i, $\sum_I$, w_i) is the function that represents the normalized mean aligned by covariance and weight. The UBM can be expressed by

$$u = \{w_i^{(u)}, m_i^{(u)}, \sum_i^{(u)} | \ I = 1,2,\ldots\ldots M\} \quad (3)$$

The speaker GMM, , can be obtained by MAP adaptation, and it has the same form as follows:

$$\lambda = \{w_i^{(\lambda)}, \ m_i^{(\lambda)}, \sum_i^{(\lambda)} | \ I = 1,2,\ldots\ldots M\} \quad (4)$$

The process of generating the GMM-supervector can be summarized in above figure. The GMM-supervector is formed by concatenating the normalized means of the Gaussian components.
Suppose we have a Gaussian mixture model universal background model (GMM UBM),

$$g(x) = \sum_{i=1}^{N} \lambda_i \, N(x; m_i,\Sigma_i) \quad (5)$$

where λ_i are the mixture weights, N() is a Guassian, and m_i and $\sum_i$ are the mean and covariance of the Guassians, respectively. We aasume diagonal covariances, $\sum$.

3. EM ALGORITHM

The most popular and well-established method for estimating the parameters of a GMM is maximum likelihood estimation[1,8,10]. Given a Gaussian mixture model[2,3,7,9], the goal is to maximize the likelihood function with respect to the parameters.

1. Initialize the means μ_k, covariances C_k and mixing coefficients Π_k, and evaluate the initial value of the log likelihood.
2. E step: Evaluate the responsibilities $\gamma_{(znk)}$ using current parameter values.
3. M step: Re-estimate the parameters μ_k^{new}, C_k^{new}, Π_k^{new} and using the current responsibilities.
4. Evaluate the log likelihood and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.

3.1 Log likelyhood Calculation :

Another quantity that plays an important role is the conditional probability of z given x Let γ(qk) denote p(qk|x). Using Baye's theorem

$$\Upsilon(q_k) \equiv P(q_k|x) = \frac{P(q_k=1)P(x|q_{k=1})}{\sum_{j=1}^{k} P(q_k=1)P(x|q_{k=1})} \quad (6)$$

$$\gamma(q_k) \equiv p(q_k \mid x) = \frac{\Pi_k N(x \mid \mu_k, c_k)}{\sum_{j=1}^{k} \pi_j N(x \mid \mu_j, c_j)} \quad (7)$$

Where Π_k is the prior probability of qk , $\gamma(q_k)$ is the responsibility that component k takes for explaining the observation x .

After the EM step the values converge i.e. they become stable. This is the end of training of speaker models. After this step unknown speaker are tested against the trained samples this is done by using "lmultiguass.m" function. Start from M initial Gaussian Models $N(\mu_k,c_k)$, k=1…..M, with equal priors set to $P(q_k|x)=1/M$.

3.2 Mathematical Background:

Estimation Step:

Compute the probability $P(q_k|X_n)$ for each data point X_n to belong to the mixture q_k.

$$p(q_k \mid x_n,\theta) = \frac{p(q_k \mid \theta).p(x_n \mid q_k,\theta)}{p(x_n \mid \theta)} \quad (8)$$

$$= \frac{p(q_k \mid \theta).p(x_n \mid q_k,\Sigma_k)}{\sum_j p(q_j \mid \theta).p(x_n \mid q_j,\Sigma_j)} \quad (9)$$

Maximization Step:

Update means

$$\mu_k^{new} = \frac{\sum_{n=1}^{T} x_n p(q_k \mid x_n,\theta)}{\sum_{n=1}^{T} p(q_k \mid x_n,\theta)} \quad (10)$$

Update Variances

$$\sum_k^{new} = \frac{\sum_{n=1}^{T} p(q_k \mid x_n,\theta)(x_n-\mu_k^{new})(x_n-\mu_k^{new})T}{\sum_{n=1}^{T} p(q_k \mid x_n,\theta)} \quad (11)$$

Update weights

$$p(q_k^{new} \mid \theta^{new}) = \frac{1}{T}\sum_{n=1}^{T} p(q_k \mid x_n,\theta) \quad (12)$$

4. PROPOSED WORK

The proposed work basically contains two broad parts, they are,

1) Designing and simulation of speaker recognition system using Gaussian super vector.
2) Testing of designed system

1) Designing of speaker recognition system using Gaussian supervector Following steps are followed for designing of speaker recognition system using Gaussian supervector:

1) For training, input utterances are loaded so that features can be easily extracted.
2) In feature extraction process, Mel frequency Cepstral coefficient (MFCC) is determined.
3) In addition to estimating GMM parameters via the EM algorithm, the parameters may also be estimated using Maximum A Posteriori (MAP) estimation. MAP estimation is used, for example, in speaker recognition applications to derive speaker model by adapting from a universal background model (UBM).
4) After these steps Gaussian supervector is generated and is saved as MFCC Gaussian supervector data .

All the above steps are shown diagrammatically below in figure 2.

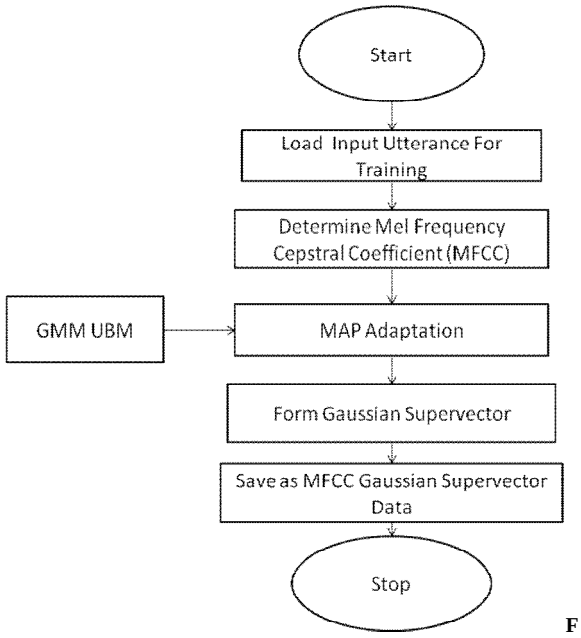

Figure 2. Design and simulation of Speaker Recognition System using Gaussian supervector.

2) Testing of designed Speaker Recognition system. In this part testing is done by extracting features of voice signal. Following steps are followed for testing :

1) Input utterances are loaded for testing so that features can be easily extracted.
2) Feature extraction is done same as training part.
3) In this step MFCC Gaussian supervector data is matched with the extracted feature of step 2.If it matches then person is known otherwise the person is unknown. It is shown in figure 3.

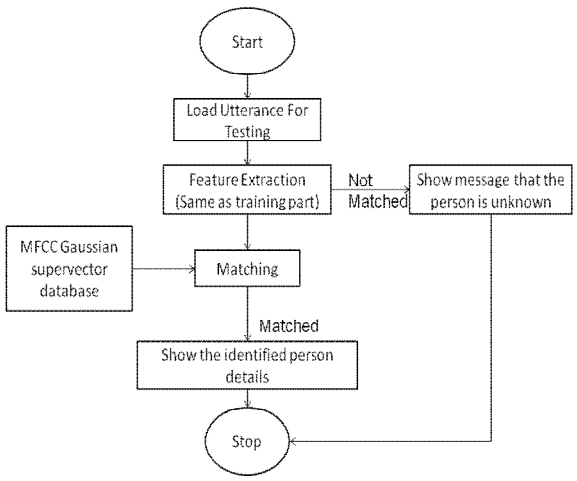

Figure 3. Testing of designed Speaker Recognition System.

5. PERFORMANCE

It is quite difficult to characterize the performance of speaker Identification systems in all applications due to the complexities and differences in the enrollment/testing scenarios. However, in this section we attempt to provide a range of performance for some cases. These numbers are not meant to indicate the best performance that can be obtained, but rather a relative ranking of some different scenarios. Some of the broad factors that can affect the performance of a speaker recognition system are:

- Speech quality: types of microphones used, ambient noise levels types of noise, compression of speech, etc.
- Speech modality: text – dependent or text – independent.
- Speech duration: amount to train and test data, temporal separation of training and testing data.
- Speaker population: number and similarity of speakers.

No one data will match the factors for all applications, but to get realistic results from the evaluation of the system , the data and design should match the target application as much as possible .Knowing performance using clean fixed text speech will indicate little about performance using free text , telephone speech . The performance of the technique has been measured by three parameters: Number of Speakers in Database, Number of Persons Tested and the % Error. Results are shown in Table 1, Table 2 and Table 3.

Table 1: Table Of Error(%) When Number Of Speakers Tested is 20

Number Of Speakers In Database	Number Of Speakers Tested	Number Of Speakers Wrongly Identified	% Error
5	20	3	15
10	20	10	50
15	20	10	50
20	20	11	55
25	20	11	55

Table 2: Table Of Error (%) When Number Of Speakers Tested is 25

Number Of Speakers In Database	Number Of Speakers Tested	Number Of Speakers Wrongly Identified	% Error
5	25	3	12
10	25	10	40
15	25	10	40
20	25	12	48
25	25	12	48

Table 3: Table Of Error (%) When Number Of Speakers Tested is 30

Number Of Speakers In Database	Number Of Speakers Tested	Number Of Speakers Wrongly Identified	% Error
5	30	3	10
10	30	12	40
15	30	12	40
20	30	14	46.6
25	30	14	46.6

6. APPLICATIONS OF SPEAKER VERIFICATION

Various areas where speaker verification is desirable are listed below:

6.1 On-site applications

On-site applications consist of all the application where the user needs to be authenticated in front of a system. Examples include access control to some facilities etc.

6.2 Remote applications

Remote applications consist of all the applications where the access to the system is made through a remote terminal inorder to secure the access to reserved services or to authenticate the user making a particular transaction (e-trade, banking transaction, etc.).

6.3 Information structuring

This application includes organizing the information in audio documents.

6.4 Games

We may implement the application in child toys and video games.

7. CONCLUSIONS AND FUTURE TRENDS

we show performance of speaker identification for text independent system on three situation .Performance is shown here as the percent error (misidentification of true speaker) for increasing speaker set sizes .One general point can be made from this plot that error rate increases with number of speaker in database . This occurs because there is more speakers to distinguish among as the speaker set size increases. With text-independent systems making commercial headway, R&D effort will shift to the more difficult issues in unconstrained situations. This includes noise conditions. There has been significant progress made in speaker identification technology and applications, but there still remain many current problems to overcome, such as noise robustness. In the current systems low level spectrum features are

used but there are many other sources of speaker information in the speech signal like idiolect (word usage), prosodic measures and other long-term signal measures which can be used. This work will be aided by the increasing use of reliable speech recognition systems for speaker recognition R&D. High-level features not only offer the potential to improve accuracy, they may also help improve robustness since they should be less susceptible to channel effects. Speaker recognition continues to be data-driven field, setting the lead among other biometrics in conducting benchmark evaluations and research on realistic data. The continued ease of collecting and making available speech from real applications means that researchers can focus on more real-world robustness issues that appear. Obtaining speech from a wide variety of handsets, channels and acoustic environments will allow examination of problem cases and development and application of new or improved compensation techniques. Currently NIST conducts annual speaker verification evaluations in which participation is open to any interested parties. There has been significant progress made in speaker recognition technology and applications, but there still remain many current problems to overcome, such as channel and noise robustness, as well as new areas to explore.

REFERENCES

8.1. Journal Article

[1] Maximum Likelihood from Incomplete Data via the EM Algorithm A. P. Dempster; N. M. Laird; D.B. Rubin Journal of the Royal Statistical Society. Series B (Methodological), Vol. 39, No. 1.(1977), pp. 1-38.

[2] Douglas A. Reynolds, Automatic speaker Recognition : Current Approaches and Future Trends MIT Lincoln Laboratory Lexington ,MA USA dar@ll.mit.edu, ICASSP2001

[3] Douglas A. Reynolds, An overview on Automatic speaker Recognition Technology. MIT Lincoln Laboratory Lexington ,MA USA dar@ll.mit.edu 2002 IEEE

[4] Fr ´ed´ eric Bimbot,1 Jean-Franc¸ois Bonastre,2 Corinne Fredouille,2 Guillaume Gravier,1 Ivan Magrin-Chagnolleau,3 SylvainMeignier,2 Teva Merlin,2 Javier Ortega-Garc´ýa,4 Dijana Petrovska-Delacr ´ etaz,5 and Douglas A. Reynolds6 " A Tutorial on Text-Independent Speaker Verification " ,EURASIP Journal on Applied Signal Processing 2004:4, 430–451 dawi Publishing Corporation

[5] A. Stolcke, E.Shriberg, L. Ferrer, S. Kajarekar, K. Sonmez, G. Tur Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA , speech recognition as feature extraction for speaker recognition , 11-13 Apr. SAFE 2007,Washington ,DC, USA

[6] Campbell ,W.,Sturim,D.. And Reynolds. D ,Support Vector Machine using GMM Supervector for Speaker Verification,IEEE Signal processing letters 13,5(may 2006)

[7] Deshak , N., And chollet , G , support vector GMMs for speaker verification , june 2006

[8] Zhenchun Lei , "Combining the Likelihood and the Kullback-Leibler Distance in Estimating the Universal Background Model for Speaker Verification Using SVM" International Conference on Pattern Recognition 2010

[9] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian

mixture speaker models," IEEE Trans. Speech, and Audio Processing , 1995

[10] Maximum a Posteriori Linear Regression for speaker recognition Xiang Zhang, Haipeng Wang , Xiang Xiao, Jianping Zhang, Yonghong Yan Think IT Speech Lab, Institute of Acoustics, Chinese Academy of Sciences, Beijing {xzhang, hpwang, xxiao, jzhang.