

SPEAKER VERIFICATION BASED ON SAUDI ACCENTED ARABIC DATABASE

Mohamed Alkanhal, Mansour Alghamdi and Zeeshan Muzaffar
King Abdulaziz City for Science and Technology
Riyadh
Saudi Arabia
{alkanhal, mghamdi, sshair}@kacst.edu.sa

ABSTRACT

Speaker verification is concerned with verifying the speaker's claimed identity. This paper reports on recent experiments we carried out for speaker verification using a Saudi accented Arabic telephone speech database with 1033 speakers. Gaussian Mixture Model was employed in these experiments. In speaker verification, users might produce two or more utterances. We show that we can reduce error rates by combining scores of these utterances.

1. INTRODUCTION

Speaker verification concerns the problem of verifying whether a given utterance has been pronounced by a claimed authorized speaker. This technique will help make it possible to implement access control by voice in many applications. In speaker verification, the claimant is requested to give a sample of speech. Depending upon the type of the security, the system might ask to give more samples of speech if the match is borderline.

There are two popular classification approaches in the area of text independent speaker verification: Generative and Discriminative. Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) are generative approaches [1], whereas Support Vector Machine (SVM) [2] and Artificial Neural Network (ANN) [3] are discriminative approaches.

Current state-of-the-art text independent speaker verification systems use GMM [4]. The reasons for popularity of the GMM as the classifier are that it is computationally inexpensive, is based on well defined statistical model, and its insensitivity to the temporal aspects of the speech for text independent speaker verification tasks [5]. To date, HMMs have not shown any advantage over GMMs in the context of text independent speaker verification tasks [5].

The SVM has been applied for speaker verification [6]. The problems with SVM are the need to search for an

appropriate kernel function for a particular application and its infelicity to handle the temporal characteristics of the speech signals [5]. ANN has also been used for speaker identification and verification [7, 8, 9]. The main advantage of ANN is its discriminative power. The main disadvantage is the need to select the optimal structure by trial and error procedures [5].

The above discussion has led us to employ GMM for our speaker verification system. This paper reports on recent experiments we carried out for speaker verification using the Saudi accented Arabic voice bank (SAAVB) [10].

This paper also investigates combining scores generated from multiple utterances to improve the performance of the system. This paper is organized as follows. Section 2 reviews the Gaussian Mixture Model for speaker verification. In Section 3, we describe SAAVB database. We describe our system in Section 4. Our experimental results are presented and discussed in Section 5. The concluding remarks are presented in Section 6.

2. BACKGROUND

2.1. Statistical Modeling

In speaker verification, we get a speech signal Y and a hypothesized speaker S . The goal of the single speaker verification is to determine whether the speech signal Y is produced by S . In order to perform this, we need first to define the following two hypotheses [5].

H_0 : Y belongs to hypothesized speaker S .

H_1 : Y does not belong to hypothesized speaker S .

The single speaker verification task can be accomplished by applying the following likelihood ratio test.

$$\left. \begin{array}{l} p(Y / H_0) \geq \theta, \text{ accept } H_0 \\ p(Y / H_1) < \theta, \text{ accept } H_1 \end{array} \right\} \quad (1)$$

where $p(Y|H_0)$ and $p(Y|H_1)$ are the likelihoods for the hypotheses H_0 and H_1 , respectively. The decision threshold for accepting or rejecting H_0 is θ . One important goal in designing the speaker verification system is to determine the technique to compute the two likelihood values.

In practice, a suitable mathematical model λ_{hyp} is used to model H_0 , which characterizes the hypothesized speaker S . In simple terms a mathematical model, say Gaussian Mixture Model (GMM), can be trained using the feature vectors extracted from the previously observed speech signals of S . This trained model λ_{hyp} , containing mean vector and covariance matrix, will represent the distribution of feature vectors for H_0 . Similarly, the model $\lambda_{\overline{hyp}}$ represents the alternative hypothesis H_1 . The likelihood ratio can then be written as:

$$\frac{p(Y | \lambda_{hyp})}{p(Y | \lambda_{\overline{hyp}})} \quad (2)$$

Usually, the logarithm of this ratio is used as:

$$\Lambda(Y) = \log p(Y | \lambda_{hyp}) - \log p(Y | \lambda_{\overline{hyp}}) \quad (3)$$

As discussed above the model λ_{hyp} (also known as client model) is well defined, but the model $\lambda_{\overline{hyp}}$ is not well defined. Different researchers have used different strategies to model alternative hypothesis. In this work, a separate model (also known as background model) for each speaker is created. Each model is trained using the features extracted from the cohort of a particular speaker S . The cohort means the set of speakers whose vocal/speech characteristics closely match to that of a particular speaker S .

2.2. Gaussian Mixture Models

Gaussian mixture models play a vital role in text independent (where there is no prior knowledge of what the speaker will say) speaker verification systems [5]. The GMM can be used as likelihood function for client and background models. Although, some other complicated likelihood functions such as those based on HMM can also be used, but to date those complicated functions have not shown any advantage over GMM in the context of text independent speaker verification tasks, like in the NIST speaker recognition evaluation [5].

Given a D -dimensional feature vector \bar{x} , the mixture density used for the likelihood function is defined as follows:

$$p(\bar{x} | \lambda) = \sum_{i=1}^M w_i p_i(\bar{x}) \quad (4)$$

The density is a weighted linear combination of M unimodal Gaussian densities $p_i(\bar{x})$ each parameterized by $D \times 1$ mean vector $\bar{\mu}_i$ and a $D \times D$ covariance matrix Σ_i :

$$p_i(\bar{x}) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(\bar{x} - \bar{\mu}_i)' \Sigma_i^{-1} (\bar{x} - \bar{\mu}_i)} \quad (5)$$

The mixture weights w_i further satisfy the constraint $\sum_{i=1}^M w_i = 1$. Collectively, a GMM is represented as $\lambda = (w_i, \bar{\mu}_i, \Sigma_i)$, $i = (1, \dots, M)$.

The maximum likelihood model parameters are estimated using the iterative Expectation-Maximization (EM) algorithm [11], provided that a collection of training vectors are available. This algorithm refines the GMM parameters iteratively to monotonically maximize the likelihood of the estimated model for the observed feature vectors i.e., for iterations k and $k+1$, $p(\bar{x} | \lambda^{k+1}) \geq p(\bar{x} | \lambda^k)$. Normally, 5 to 10 iterations are enough for parameter convergence.

3. SAAVB DATABASE

The Saudi Accented Arabic Voice Bank (SAAVB) is a telephony speech database that was collected by King Abdulaziz City for Science and Technology (KACST) during 2002 and 2003 [10]. This data was acquired from 1033 native speakers of Arabic with Saudi accent (51% males and 49% females). Around 51% of those speakers are aged 16 to 30 years and 49% of them are above 30 years old. 70% of the database was recorded over the Saudi mobile network while the remaining 30% was acquired through the fixed-line network. Tables 1 and 2, respectively, show the SAAVB recording environments for mobile and fixed-line networks.

Each speaker read 59 prompts that consist of: numbers, phonetically rich words, sentences and pronunciation of the Arabic and English alphabets. The average number of words in each prompt is 5 words.

Table 1. Recording environments for mobile network

Recording environment	%
Quiet	34.76
Noisy	34.76
Moving vehicle	30.48

Table 2. Recording environments for fixed-line network.

Recording environment	%
Quiet	75.32%
Noisy	24.68%

The duration of the total recorded speech in SAAVB is 96.37 hours distributed among 60947 audio files (1033 speakers x 59 audio files). This means that the average duration for each speaker is 5.60 minutes and the average duration of each audio file is 5.70 seconds. The database was digitized at 8000 Hz sampling rate with A/D conversion precision of 8 bits.

The total size of SAAVB is 2.59 GBytes. It consists of 1033 directories with 183,518 files. Every directory represents a speaker and contains 178 files distributed as follows:

1. text document that has all the information needed about the speaker (gender, telephone type, age and acoustic environment).
2. 59 text files of the prompts.
3. 59 text files of the speech transcription.

4. SYSTEM DESCRIPTION

The main idea in our speaker verification system is to develop client and background models for various speakers in the SAAVB database. We have used GMMs to create these client and background models.

The following steps were used to develop the speaker verification system:

1. Create a separate GMM based client model for each speaker. Each model is then trained using features extracted from the speech signals of the particular speaker in the training database.
2. The MFCCs and energy component are used as features.
3. In order to create background list for each speaker we employed a GMM clustering algorithm proposed in [12]. This algorithm makes use of Kullback-Leibler distance as a metric to determine the closeness of GMMs.
4. The outcomes of the clustering algorithm are various clusters and the corresponding GMM models that are trained using the combined features of the members of particular cluster.
5. The cluster information can then be used to create background list as well as the background model for each speaker. This background model is actually a GMM trained using the combined features of all the background speakers for a particular speaker.
6. Although one can use the client model and a single background model in order to verify a particular speaker, we have used the client

models of all the background speakers as the background models for that particular speaker. This helps in normalization during verification. We have made use of Tnorm normalization as mentioned in [13].

7. In order to make decision we simply use the log of the likelihood ratio, as described earlier, of the client model and the background model.
8. If the ratio is greater than a certain threshold then the speaker is declared legitimate.

5. RESULTS

Silence was removed from the speech signals, since it is not speaker specific and might act like a noise, affecting the speaker verification performance [14]. A 30-ms Hamming window was applied to the speech signals every 10-ms yielding speech frames. Each frame was processed using the Mel-Cepstral analysis to obtain 12 MFCCs and one energy term. These features were augmented by the first and second order derivatives yielding 39 dimensional vectors.

We assume each claimant is asked to utter multiple sentences during each verification session. Thus, for each client and imposter, we applied the score fusion. We have conducted various experiments by making decisions on single utterances and by averaging scores generated by multiple utterances, which means that the fusion weights are equal for all scores [15].

In speaker verification there are two types of errors that can be made: a false acceptance where an imposter is incorrectly authenticated and a false rejection where a user is incorrectly identified as an imposter. We have computed the equal error rate (EER) with equal costs for a false acceptance and a false rejection. We have used one global threshold for the decision making. After obtaining verification scores for different utterances, we combined them to obtain better verification results

Fig 1 shows the speaker detection performance for the single utterance based system and the score averaging approach (with three and five utterances). In this figure, we can see the tradeoffs of the two errors. It is shown clearly that score fusion is able to reduce error rates.

The EERs for single utterances and the score averaging approach (with 3 and 5 utterances) are shown in Table 3. We can see that the average EER achieved by single utterances is 5.90%. When applying the score averaging for three and five utterances, the average EER reduces to 5.40% and 4.58%, respectively.

Table 3. Average EER for single utterance based system and for the score averaging approach (for three and five utterances).

Number of utterances	Average EER %
1	5.90
3	5.40
5	4.58

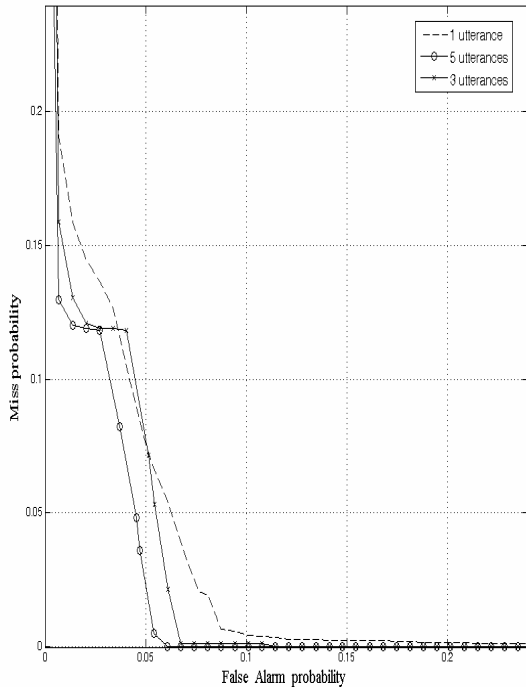


Figure 1. Speaker detection performance for single utterance and for score averaging of three and five utterances using the Gaussian mixture model (GMM) classifier. It is shown that combining scores of multiple utterances can improve the performance of the system.

6. CONCLUSION

In this paper, we presented a speaker verification system for Saudi accented Arabic telephone speech database. Results based on 1033 speakers show that the overall performance of a speaker verification system can be improved by combining scores of multiple utterances. We are currently extending the fusion method using some other combination rules.

ACKNOWLEDGEMENTS

This work is supported by King Abdulaziz City for Science and Technology under project number CI-26-01.

REFERENCES

[1] G.R. Doddington, M.A. Przybycki, A.F. Martin, and D.A. Reynolds, "The NIST speaker recognition evaluation -

Overview, methodology, systems, results, perspective," *Speech Communication*, vol. 31, pp. 225-254, 2000.

[2] V. N. Vapnick, *The nature of Statistical Learning Theory*, Springer, 1995.

[3] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[4] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech communication*, vol. 17, pp. 91-108, 1995.

[5] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Margin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska-Delacretaz, and D.-A. Reynolds, "A Tutorial on Text-Independent Speaker Verification", *EURASIP Journal on Applied Signal Processing*, vol. 4, pp: 430-451, 2004.

[6] V. Wan, and S. Renals, "SVMSVM: Support Vector Machine Speaker Verification Methodology", in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vol. 2, pp. 221-224, 2003.

[7] J. Oglesby, and S. Mason, "Optimization of neural models for speaker identification", in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, pp. 261-264, 1990.

[8] Y. Bennani, and P. Gallinari, "Connectionist approaches for automatic speaker recognition," in *Proc. the 1st ESCA Work. Automatic Speaker Recognition, Identification, Verification*, pp. 95-102, 1994.

[9] K. R. Farrell, R. Mammone, and K. Assaleh, "Speaker recognition using neural networks and conventional classifiers", *IEEE Trans. Speech, Audio Processing*, vol. 2, no. 1, pp. 194-205, 1994.

[10] M. Alghamdi, F. Alhargan, M. Alkanhal, A. Alkhairi and M. Aldusuqi, *Saudi Accented Arabic Voice Bank*, Final report, Computer and Electronics Research Institute, King Abdulaziz City for Science and technology, Riyadh, Saudi Arabia, 2003.

[11] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1-8, 1997.

[12] B. Sunt, W. Lid, and Q. Zhon, "Hierarchical Speaker Identification using Speaker Clustering", *Int. Conf. natural Language processing and knowledge engineering*, pp. 299-304, 2003.

[13] M. Carey, R. Auckenthaler, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42-54, 2000.

[14] R. Hu and R. Damper, "Fusion of two classifiers for speaker identification: removing and not removing silence," in *Proc. the 8th International conference on Information fusion*, vol 1, pp. 429-436, 2006.

[15] N. Poh, S. Bengio, and J. Korczak, "A Multi-sample Multi-source Model for Biometric Authentication," in *Proc. IEEE 12th Workshop on Neural Networks for Signal Processing*, pp. 375-384, 2002.