

Speaker Indexing In Audio Archives

Using Test Utterance Gaussian Mixture Modeling

Hagai Aronowitz¹, David Burshtein² and Amihod Amir^{1,3}

¹Department of Computer Science, Bar-Ilan University, Israel

²School of Electrical Engineering, Tel-Aviv University, Israel

³College of Computing, Georgia Tech, USA

aronowc@cs.biu.ac.il, burstyn@eng.tau.ac.il, amir@cs.biu.ac.il

Abstract

Speaker Indexing has recently emerged as an important task due to the rapidly growing volume of audio archives. Current filtration techniques still suffer from problems both in accuracy and efficiency. The major reason for the drawbacks of existing solutions is the use of inaccurate anchor models. The contribution of this paper is two-fold. On the theoretical side, a new method is developed for simulating GMM scoring. This enables to fit a GMM not only to every target speaker but also to every test utterance, and then compute the likelihood of the test call using these GMMs instead of using the original data. The second contribution of this paper is in harnessing this GMM simulation to achieve very efficient speaker indexing in terms of both search time and index size. Results on the SPIDRE corpus show that our approach maintains the accuracy of the conventional GMM algorithm.

1. Introduction

Indexing large audio archives has emerged recently [5, 6] as an important research topic as large audio archives now exist. The goal of speaker indexing is to divide the speaker recognition process into 2 stages. The first stage is a pre-processing phase which is usually done on-line as audio is inserted into the archive. In this stage there is no knowledge about the target speakers. The goal of the pre-processing stage is to do all possible pre-calculations in order to make the search as efficient as possible when a query is presented. The second stage is activated when a target speaker query is presented. In this stage the pre-calculations of the first stage are used.

Previous research such as [7] suggests projecting each utterance into a speaker space defined by anchor models which are a set of non-target speaker models. Each utterance is represented by a vector of distances between the utterance and each anchor model. This representation is calculated in the pre-processing phase. In the query phase, the target speaker data is projected to the same speaker space and the speaker space representation of each utterance in the archive is compared to the target speaker vector, using a distance measure such as Euclidean distance. The disadvantage of this approach is that it is intuitively suboptimal (otherwise, it would replace the Gaussian mixture model (GMM) [1, 2] approach and wouldn't be limited to speaker indexing). Indeed, the EER reported in [7] is almost tripled when using anchor models instead of conventional GMM scoring. This disadvantage was handled in [7] by cascading the anchor

model indexing system and the GMM recognition system thus first filtering efficiently most of the archive and then rescored in order to improve accuracy. Nevertheless, the cascaded system described in [7] failed to obtain accurate performance for speaker misdetection probability lower than 50%. Another drawback of the cascade approach is that sometimes the archive is not accessible for the search system either because it is too expensive to access the audio archive, or because the audio itself was deleted from the archive because of lack of available storage resources (the information that a certain speaker was speaking in a certain utterance may be beneficial even if the audio no longer exists, for example for law enforcement systems). Therefore, it may be important to be able to achieve accurate search with low time and memory complexity using only an index file and not the raw audio.

Our suggested approach for speaker indexing is by harnessing the GMM to this task. GMM has been the state-of-the-art algorithm for this task for many years. The GMM algorithm calculates the log-likelihood of a test utterance given a target speaker by fitting a parametric model to the target training data and computing the average log-likelihood of the test utterance feature vectors assuming independence between frames. Analyzing the GMM algorithm shows asymmetry between the target training data and the test call. This asymmetry seems to be not optimal: if a Gaussian mixture model can model robustly the distribution of acoustic frames, why not use it to represent robustly the test utterance?

In [3] both target speakers and test utterances were treated symmetrically by being modeled by a covariance matrix. The distance between a target speaker and a test utterance was also defined as a symmetric function of the target model and test utterance model. In [4] cross likelihood ratio was calculated between the GMM representing a target speaker and a GMM representing a test utterance.

Therefore, the motivation for representing a test utterance by a GMM is that this representation is robust and smooth. In fact, the process of GMM fitting exploits a-priori knowledge about the test utterance - the smoothness of the distribution. Using universal background model (UBM) MAP-adaptation for fitting the GMM exploits additional a-priori knowledge. Our speaker recognition algorithm fits a GMM for every test utterance in the indexing phase (stage 1), and calculates the likelihood (stage 2) by using only the GMM of the target speaker, and the GMM of a test utterance. To our knowledge, this is the first time that a simulation of a GMM score has appeared in the literature that uses a GMM fitted to the test utterance rather than the test utterance itself. This novel contribution is a key to our efficient yet accurate algorithm.

The organization of this paper is as follows: the proposed speaker recognition system is presented in Section 2. Section 3 describes the experimental corpus, the experiments and the results for the speaker recognition. Section 4 describes the speaker indexing algorithm and analyzes its efficiency. Finally, section 5 presents conclusions and ongoing work.

2. Simulating GMM scoring

In this section we describe the proposed speaker recognition algorithm. Our goal is to simulate the calculation of a GMM score without using the test utterance data but using only a GMM fitted to the test utterance.

2.1. Definition of the GMM score

The log-likelihood of a test utterance $X = x_1, \dots, x_n$ given a target speaker GMM Q is usually normalized by some normalization log-likelihood (UBM log-likelihood, cohort log-likelihood, etc.) and divided by the length of the utterance. This process is summarized by equation (1):

$$\text{score}(X|Q) = \frac{LL(X|Q) - LL(X|\text{norm models})}{n} \quad (1)$$

Equation (1) shows that the GMM score is composed of a target-speaker dependent component – the average log-likelihood of the utterance given the speaker model ($LL(X|Q)/n$) and a target-speaker independent component – the average log-likelihood of the utterance given the normalization models ($LL(X|\text{norm-models})/n$). For simplicity, the rest of this paper will focus on a single normalization model, the UBM, but the same techniques can be trivially used for other normalization models such as cohort models. The GMM model assumes independence between frames. Therefore, the log-likelihood of X given Q is calculated in equation (2):

$$\frac{1}{n} LL(X|Q) = \frac{1}{n} \sum_{i=1}^n \log(\Pr(x_i|Q)) \quad (2)$$

2.2. GMM scoring using a model for the test utterance

The vectors x_1, \dots, x_n of the test utterance are acoustic observation vectors generated by a stochastic process. Let us assume that the true distribution of which the vectors x_1, \dots, x_n were generated by is P . The average log-likelihood of an utterance Y of asymptotically infinite length $|Y|$ generated by the distribution P is given in equation (3):

$$\frac{1}{n} LL(Y|Q) = \frac{1}{|Y|} \sum_{i=1}^{|Y|} \log(\Pr(y_i|Q)) \quad (3)$$

$$\xrightarrow[|Y| \rightarrow \infty]{x} \int \Pr(x|P) \log(\Pr(x|Q)) dx$$

The result of equation (3) is that the log-likelihood of a test utterance given distribution Q is a random variable that asymptotically converges to an integral of a function of distributions Q and P . In order to use equation (3) we have to know the true distribution P and we have to calculate the integral.

2.3. Estimation of distribution P

We assume that the test utterance is generated using a true distribution P . Therefore, P should be estimated by the same methods that distribution Q is estimated from the training data of the target speaker, i.e. by fitting a GMM, though the order of the model may be tuned to the length of the test utterance.

$$2.4. \text{ Calculation of } \int_x \Pr(x|P) \log(\Pr(x|Q)) dx$$

Definitions:

- w_i^P, w_j^Q : The weight of the $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q.
- μ_i^P, μ_j^Q : The mean vector of the $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q.
- $\mu_{i,d}^P, \mu_{j,d}^Q$: The d^{th} coordinate of the mean vector of the $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q.
- σ_i^P, σ_j^Q : The standard deviation vector of the $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q (assuming diagonal covariance matrix).
- $\sigma_{i,d}^P, \sigma_{j,d}^Q$: The d^{th} coordinate of the standard deviation vector of the $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q (assuming diagonal covariance matrix).
- P^i, Q^j : The $i^{\text{th}}/j^{\text{th}}$ Gaussian of distribution P/Q.
- $N(x|\mu, \sigma)$: The probability density of a vector x given a normal distribution with mean vector μ and standard deviation vector σ (assuming diagonal covariance matrix).
- n_g^P, n_g^Q : The number of Gaussians of distribution P/Q.
- dim : The dimension of the acoustic vector space.

Distribution P is a GMM and is defined in equation (4):

$$\Pr(x|P) = \sum_{i=1}^{n_g^P} w_i^P \Pr(x|P_i) \quad (4)$$

Using equation (4) and exploiting the linearity of the integral and the mixture model we get:

$$\int_x \Pr(x|P) \log(\Pr(x|Q)) dx = \sum_{i=1}^{n_g^P} w_i^P \int_x \Pr(x|P_i) \log(\Pr(x|Q)) dx \quad (5)$$

In order to get a closed form solution for the integral in equation (5) we have to use an approximation. Equation (6) presents an inequality that is true for every Gaussian j therefore we have n_g^Q closed form lower bounds for the integral (for every Gaussian j we get a possibly different lower bound).

$$\begin{aligned}
& \int_x \Pr(x|P_i) \log(\Pr(x|Q)) dx = \\
& = \int_x N(x | \mu_i^P, \sigma_i^P) \log \left[\sum_{j=1}^{n_Q} w_j^Q \times N(x | \mu_j^Q, \sigma_j^Q) \right] dx \\
& \geq \int_x N(x | \mu_i^P, \sigma_i^P) \log \left[w_j^Q \times N(x | \mu_j^Q, \sigma_j^Q) \right] dx \quad (6) \\
& = \log w_j^Q - \sum_{d=1}^{\dim} \frac{(\mu_{i,d}^P - \mu_{j,d}^Q)^2}{2\sigma_{j,d}^Q} - \sum_{d=1}^{\dim} \log \sigma_{j,d}^Q \\
& \quad - \frac{1}{2} \sum_{d=1}^{\dim} \left(\frac{\sigma_{i,d}^P}{\sigma_{j,d}^Q} \right)^2 - \frac{\dim}{2} \log 2\pi
\end{aligned}$$

The tightest lower bound is received by setting j to j_opt_i which is defined in equation (7):

$$j_opt_i = \arg \max_j \left\{ \begin{array}{l} \log w_j^Q - \sum_{d=1}^{\dim} \frac{(\mu_{i,d}^P - \mu_{j,d}^Q)^2}{2\sigma_{j,d}^Q} \\ - \sum_{d=1}^{\dim} \log \sigma_{j,d}^Q - \frac{1}{2} \sum_{d=1}^{\dim} \left(\frac{\sigma_{i,d}^P}{\sigma_{j,d}^Q} \right)^2 \end{array} \right\} \quad (7)$$

The approximation we use in this paper is taking the tightest lower bound defined by equations (6, 7) as an estimate to the integral $\int_x \Pr(x|P_i) \log(\Pr(x|Q)) dx$.

2.5. Speeding up Calculation of $\int_x \Pr(x|P) \log(\Pr(x|Q)) dx$

The complexity of approximating the integral is $O(g^2d)$: for every Gaussian of P the closest Gaussian (according to equation (7)) in Q must be found. This search can be accelerated without any notable loss in accuracy by exploiting the fact that both P and Q are adapted from the same UBM. Before indexing phase, the asymmetric distance between each pair of Gaussians from the UBM is computed according to equation (7). The set of all distances is sorted and a distance threshold is set according to a small percentage (ϵ) of the set. For each Gaussian i only Gaussians that are closer than the threshold are stored in a Gaussian specific list L_i . In search phase, when searching for the closest Gaussian for P_i in Q , only the Gaussians in the list L_i are examined. This suboptimal calculation of the approximation improves the time complexity to $O(\epsilon g^2d)$.

2.6. Global variance models

Global variance GMM models are GMM models with the same variance matrix shared among all Gaussians and all speakers. Using global variance GMMs has the advantages of lower time and memory complexity and also improves robustness when training data is sparse. The reduced modeling power of using a global variance can be compensated by moderately increasing the number of Gaussians. The robustness issue may be especially important when modeling

short test utterances. Applying the Global variance assumption to equations (6, 7) results in much simpler equations (8, 9):

$$\int_x \Pr(x|P_i) \log(\Pr(x|Q)) dx \geq \log w_j^Q - \sum_{d=1}^{\dim} \frac{(\mu_{i,d}^P - \mu_{j,d}^Q)^2}{2\sigma_d^2} + C \quad (8)$$

In (9) C is a speaker independent constant.

$$j_opt_i = \arg \max_j \left\{ \log w_j^Q - \sum_{d=1}^{\dim} \frac{(\mu_{i,d}^P - \mu_{j,d}^Q)^2}{2\sigma_d^2} \right\} \quad (9)$$

3. Experimental results

3.1. The SPIDRE corpus

Experiments were conducted on the SPIDRE corpus [8] which is a subset of the Switchboard-I corpus. The SPIDRE consists of 45 target speakers, four conversations per speaker, and 100 2 sided non-target conversations. All conversations are about 5 minutes long and are all from land-line phones with mixed handsets. The SPIDRE corpus is manually transcribed. The 100 non-target conversations were divided to the following subsets: fifty two-sided conversations were used as training and development data, and the other fifty two-sided conversations were used as test data. The four target conversations per speaker were divided randomly to two training conversations and two testing conversations, therefore some of the tests are in matched handset condition and some are in mismatched handset condition. The second side of the training target conversations was used as additional development data, and the second side of the testing target conversations was used as additional non-target testing data.

3.2. The baseline GMM system

The baseline GMM system in this paper was inspired by the GMM-UBM system described in [1, 2]. The front-end of the recognizer consists of calculation of Mel-frequency cepstrum coefficients (MFCC) according to the ETSI standard [9]. An energy based voice activity detector is used to locate and remove non-speech segments and the cepstral mean of the speech segments is calculated and subtracted. The final feature set is 13 cepstral coefficients + 13 delta cepstral coefficients extracted every 10ms using a 20ms window. A gender independent UBM was trained using 100 non-target conversation sides (about 8 hours of speech + non-speech). Target speakers were trained using MAP adaptation. Several model orders were evaluated – 512, 1024 and 2048 Gaussians. Both Gaussian and speaker dependent diagonal variance matrix and global diagonal variance matrix GMMs were evaluated. A fast scoring technique was used in which only the top 5 highest scoring Gaussians are rescored using the target models [2]. In the verification stage, the log likelihood of each conversation side given a target speaker is divided by the length of the conversation and normalized by the UBM score. The resulting score is then normalized using z-norm [1].

The DET curve of the GMM system with 1024 Gaussians is presented in Figure 1. The EER of the GMM system is 9.6%.

3.3. Accuracy of the GMM simulation system

The DET curve of the GMM simulation system with 1024 Gaussians is presented in Figure 1. It can be seen that the GMM simulation system performs practically the same as the GMM system. The EER of the GMM simulation system is 9.4%. Results for 512 and 2048 Gaussians show the same similarity between both systems.

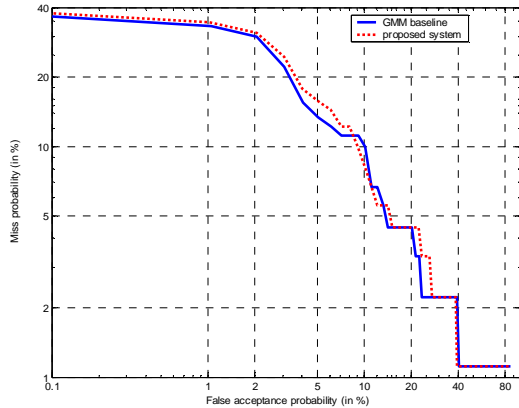


Figure 1: DET curve comparing the baseline GMM system to the GMM simulation system.

3.4. Gaussian pruning in the GMM simulation system

Experiments were done in order to find an optimal value for ϵ . For $\epsilon=3\%$ no degradation in performance was found on any experiment. For $\epsilon=1\%$ negligible degradation in performance was found on some experiments.

4. Speaker indexing

A speaker indexing system can be built using the GMM simulation algorithm. The indexing system can be measured in terms of accuracy, time complexity in indexing phase, time complexity of search phase, and the size of the index. Tables 1,2 show the search phase time complexity and index size of the GMM simulation system compared to the GMM based indexing system. In tables 1,2 g is the number of Gaussians (1024), d is the acoustic space dimension (26), n is the mean net size of a test utterance (6000 frames), ϵ is the pruning factor speedup (0.03) and c is the complexity of the ETSI front-end per frame.

	Time complexity	Time in practice
Baseline (GMM)	$O(gnd + cn)$	100.0%
GMM simulation	$O(g^2d)$	26.5%
GMM simulation + Gaussian pruning	$O(\epsilon g^2d)$	0.8%

Table 1: Search phase time complexity per test utterance of the GMM and simulated GMM indexing systems.

	Index size	Index size in KB
Baseline (GMM)	80n	1500
GMM simulation	4gd	100

Table 2: Index size per test utterance for the GMM and for the simulated GMM indexing systems.

5. Conclusions

In this paper we have presented the GMM simulation algorithm which is a method to simulate the conventional GMM scoring algorithm in a distributed way suitable for speaker indexing. A speaker indexing system based of the GMM simulation algorithm is as accurate as one based on the conventional GMM algorithm and is much faster and requires only 1/15 of the storage.

The focus of our ongoing research is reducing the size of the index and obtaining sub-linear time complexity for the search phase.

6. References

- [1] Reynolds, D. A., "Comparison of background normalization methods for text-independent speaker verification", in Proc. *Eurospeech*, pp.963-966, 1997.
- [2] McLaughlin, J., Reynolds, D. A., and Gleason, T., "A study of computation speed-ups of the GMM-UBM speaker recognition system", in Proc. *Eurospeech*, pp.1215-1218, 1999.
- [3] Schmidt M., Gish H., and Mielke A., "Covariance estimation methods for channel robust text-independent speaker identification". In Proc. *ICASSP*, pp. 333-336, 1995.
- [4] Tsai W. H., Chang W. W., Chu Y. C., and Huang C. S., "Explicit exploitation of stochastic characteristics of test utterance for text-independent speaker identification", in Proc. *Eurospeech*, pp. 771-774, 2001.
- [5] Foote J., "An overview of audio information retrieval", *ACM Multimedia Systems*, 7:2--10, 1999.
- [6] Chagolleau I. M. and Vallès N. P., "Audio indexing: What has been accomplished and the road ahead", in *JCIS*, pp. 911-914, 2002.
- [7] Sturim, D. E., Reynolds, D. A., Singer, "Speaker indexing in large audio databases using anchor models", in Proc. *ICASSP*, pp. 429-432, 2001.
- [8] Linguistic Data Consortium, SPIDRE documentation file, http://www ldc.upenn.edu/Catalog/readme_files/spidre_readme.html
- [9] "Speech processing, transmission and quality aspects (stq); distributed speech recognition; front-end feature extraction algorithm; compression algorithms," ETSI Standard: ETSI-ES-201-108-v1.1.2, 2000, <http://www.etsi.org/stq>.