Speaker Identication Using Discriminative Learning of Large Margin GMM

Khalid Daoudi, Reda Jourani, Régine André-Obrecht, Driss Aboutajdine

Speaker Identification Using Discriminative Learning of Large Margin GMM

Khalid Daoudi [1], Reda Jourani [2,3], Régine André-Obrecht [2], and Driss Aboutajdine [3]

[1] GeoStat Group, INRIA Bordeaux-Sud Ouest, Talence. France
[2] SAMoVA Group, IRIT - Univ. Paul Sabatier, Toulouse, France
[3] Laboratoire LRIT. Faculty of Sciences, Mohammed 5 Agdal Univ., Rabat, Morocco
khalid.daoudi@inria.fr,{jourani,obrecht}@irit.fr,aboutaj@fsr.ac.ma

Abstract. Gaussian mixture models (GMM) have been widely and successfully used in speaker recognition during the last decades. They are generally trained using the generative criterion of maximum likelihood estimation. In an earlier work, we proposed an algorithm for discriminative training of GMM with diagonal covariances under a large margin criterion. In this paper, we present a new version of this algorithm which has the major advantage of being computationally highly efficient, thus well suited to handle large scale databases. We evaluate our fast algorithm in a Symmetrical Factor Analysis compensation scheme. We carry out a full NIST speaker identification task using NIST-SRE'2006 data. The results show that our system outperforms the traditional discriminative approach of SVM-GMM supervectors. A 3.5% speaker identification rate improvement is achieved.

Key words: Large margin training, Gaussian mixture models, Discriminative learning, Speaker recognition, Session variability modeling

1 Introduction

Most of state-of-the-art speaker recognition systems rely on the generative training of Gaussian Mixture Models (GMM) using maximum likelihood estimation and maximum a posteriori estimation (MAP) [1]. This generative training estimates the feature distribution within each speaker. In contrast, the discriminative training approaches model the boundary between speakers [2, 3], thus generally leading to better performances than generative methods. For instance, Support Vector Machines (SVM) combined with GMM supervectors are among state-of-the-art approaches in speaker verification [4, 5].

In speaker recognition applications, mismatch between the training and testing conditions can decrease considerably the performances. The inter-session variability, that is the variability among recordings of a given speaker, remains the most challenging problem to solve. The Factor Analysis techniques [6, 7], e.g., Symmetrical Factor Analysis (SFA) [8], were proposed to address that problem in GMM based systems. While the Nuisance Attribute Projection (NAP) [9] compensation technique is designed for SVM based systems.

Recently a new discriminative approach for multiway classification has been proposed, the Large Margin Gaussian mixture models (LM-GMM) [10]. The latter have the same advantage as SVM in term of the convexity of the optimization problem to solve. However they differ from SVM because they draw nonlinear class boundaries directly in the input space. While LM-GMM have been used in speech recognition, they have not been used in speaker recognition (to the best of our knowledge). In an earlier work [11], we proposed a simplified version of LM-GMM which exploit the fact that traditional GMM based speaker recognition systems use diagonal covariances and only the mean vectors are MAP adapted. We then applied this simplified version to a "small" speaker identification task. While the resulting training algorithm is more efficient than the original one, we found however that it is still not efficient enough to process large databases such as in NIST Speaker Recognition Evaluation (NIST-SRE) campaigns (*http://www.itl.nist.gov/iad/mig//tests/sre/*).

In order to address this problem, we propose in this paper a new approach for fast training of Large-Margin GMM which allow efficient processing in large scale applications. To do so, we exploit the fact that in general not all the components of the GMM are involved in the decision process, but only the k-best scoring components. We also exploit the property of correspondence between the MAP adapted GMM mixtures and the Universal Background Model mixtures [1]. In order to show the effectiveness of the new algorithm, we carry out a full NIST speaker identification task using NIST-SRE'2006 (core condition) data. We evaluate our fast algorithm in a Symmetrical Factor Analysis (SFA) compensation scheme, and we compare it with the NAP compensated GMM supervector Linear Kernel system (GSL-NAP) [5]. The results show that our Large Margin compensated GMM outperform the state-of-the-art discriminative approach GSL-NAP.

The paper is organized as follows. After an overview on Large-Margin GMM training with diagonal covariances in section 2, we describe our new fast training algorithm in section 3. The GSL-NAP system and SFA are then described in sections 4 and 5, respectively. Experimental results are reported in section 6.

2 Overview on Large Margin GMM with Diagonal Covariances (LM-dGMM)

In this section we start by recalling the original Large Margin GMM training algorithm developed in [10]. We then recall the simplified version of this algorithm that we introduced in [11].

In Large Margin GMM [10], each class c is modeled by a mixture of ellipsoids in the D-dimensional input space. The m^{th} ellipsoid of the class c is parameterized by a centroid vector μ_{cm}, a positive semidefinite (orientation) matrix Ψ_{cm} and a nonnegative scalar offset $\theta_{cm} \geq 0$. These parameters are then collected into a single enlarged matrix Φ_{cm}:

$$\Phi_{cm} = \begin{pmatrix} \Psi_{cm} & -\Psi_{cm}\mu_{cm} \\ -\mu_{cm}^T\Psi_{cm} & \mu_{cm}^T\Psi_{cm}\mu_{cm} + \theta_{cm} \end{pmatrix}. \quad (1)$$

A GMM is first fit to each class using maximum likelihood estimation. Let $\{o_{nt}\}_{t=1}^{T_n}$ ($o_{nt} \in \mathcal{R}^D$) be the T_n feature vectors of the n^{th} segment (i.e. n^{th} speaker training data). Then, for each o_{nt} belonging to the class y_n, $y_n \in \{1, 2, ..., C\}$ where C is the total number of classes, we determine the index m_{nt} of the Gaussian component of the GMM modeling the class y_n which has the highest posterior probability. This index is called *proxy label*. The training algorithm aims to find matrices Φ_{cm} such that "all" examples are correctly classified by at least one margin unit, leading to the LM-GMM criterion:

$$\forall c \neq y_n,\ \forall m, \qquad z_{nt}^T \Phi_{cm} z_{nt}\ \geq 1 + z_{nt}^T \Phi_{y_n m_{nt}} z_{nt}, \tag{2}$$

where $z_{nt} = [o_{nt}\ \ 1]^T$.

In speaker recognition, most of state-of-the art systems use diagonal covariances GMM. In these GMM based speaker recognition systems, a speaker-independent *world model* or *Universal Background Model* (UBM) is first trained with the EM algorithm. When enrolling a new speaker to the system, the parameters of the UBM are adapted to the feature distribution of the new speaker. It is possible to adapt all the parameters, or only some of them from the background model. Traditionally, in the GMM-UBM approach, the target speaker GMM is derived from the UBM model by updating only the mean parameters using a *maximum a posteriori* (MAP) algorithm [1].

Making use of this assumption of diagonal covariances, we proposed in [11] a simplified algorithm to learn GMM with a large margin criterion. This algorithm has the advantage of being more efficient than the original LM-GMM one [10] while it still yielded similar or better performances on a speaker identification task. In our Large Margin diagonal GMM (LM-dGMM) [11], each class (speaker) c is initially modeled by a GMM with M diagonal mixtures (trained by MAP adaptation of the UBM in the setting of speaker recognition). For each class c, the m^{th} Gaussian is parameterized by a mean vector μ_{cm}, a diagonal covariance matrix $\Sigma_m = diag(\sigma_{m1}^2, ..., \sigma_{mD}^2)$, and the scalar factor θ_m which corresponds to the weight of the Gaussian. For each example o_{nt}, the goal of the training algorithm is now to force the log-likelihood of its proxy label Gaussian m_{nt} to be at least one unit greater than the log-likelihood of each Gaussian component of all competing classes. That is, given the training examples $\{(o_{nt}, y_n, m_{nt})\}_{n=1}^N$, we seek mean vectors μ_{cm} which satisfy the LM-dGMM criterion:

$$\forall c \neq y_n,\ \forall m, \qquad d(o_{nt}, \mu_{cm}) + \theta_m \geq 1 + d(o_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}}, \tag{3}$$

where $d(o_{nt}, \mu_{cm}) = \sum_{i=1}^{D} \frac{(o_{nti} - \mu_{cmi})^2}{2\sigma_{mi}^2}$. Afterward, these M constraints are fold into a single one using the softmax inequality $min_m\ a_m \geq -log \sum_m e^{-a_m}$. The segment-based LM-dGMM criterion becomes thus:

$$\forall c \neq y_n, \quad \frac{1}{T_n} \sum_{t=1}^{T_n} -log \sum_{m=1}^{M} e^{(-d(o_{nt}, \mu_{cm}) - \theta_m)} \geq 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} d(o_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}}. \tag{4}$$

Letting $[f]_+ = max(0, f)$ denote the so-called *hinge* function, the loss function to minimize for LM-dGMM is then given by:

$$Ł = \sum_{n=1}^{N} \sum_{c \neq y_n} \left[1 + \frac{1}{T_n} \sum_{t=1}^{T_n} \left(d(o_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}} + log \sum_{m=1}^{M} e^{(-d(o_{nt}, \mu_{cm}) - \theta_m)} \right) \right]_+ . \tag{5}$$

3 LM-*d*GMM Training with *k*-best Gaussians

3.1 Description of the New LM-dGMM Training Algorithm

Despite the fact that our LM-dGMM is computationally much faster than the original LM-GMM of [10], we still encountered efficiency problems when dealing with high number of Gaussian mixtures. In order to develop a fast training algorithm which could be used in large scale applications such as NIST-SRE, we propose to drastically reduce the number of constraints to satisfy in (4). By doing so, we would drastically reduce the computational complexity of the loss function and its gradient. To achieve this goal we propose to use another property of state-of-the-art GMM systems, that is, decision is not made upon all mixture components but only using the k-best scoring Gaussians. In other words, for each o_n and each class c, instead of summing over the M mixtures in the left side of (4), we would sum only over the k Gaussians with the highest posterior probabilities selected using the GMM of class c. In order to further improve efficiency and reduce memory requirement, we exploit the property reported in [1] about correspondence between MAP adapted GMM mixtures and UBM mixtures. We use the UBM to select one unique set S_{nt} of k-best Gaussian components per frame o_{nt}, instead of $(C-1)$ sets. This leads to a $(C-1)$ times faster and less memory consuming selection. More precisely, we now seek mean vectors μ_{cm} that satisfy the large margin constraints in (6):

$$\forall c \neq y_n, \quad \frac{1}{T_n} \sum_{t=1}^{T_n} -log \sum_{m \in S_{nt}} e^{(-d(o_{nt}, \mu_{cm}) - \theta_m)} \geq 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} d(o_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}}. \tag{6}$$

The resulting loss function expression is straightforward. During test, we use again the same principle to achieve fast scoring. Given a test segment of T frames, for each test frame x_t we use the UBM to select the set E_t of k-best scoring proxy labels and compute the LM-dGMM likelihoods using only these k labels. The decision rule is thus given as:

$$y = argmin_c \left\{ \sum_{t=1}^{T} -log \sum_{m \in E_t} e^{(-d(o_t, \mu_{cm}) - \theta_m)} \right\}. \tag{7}$$

3.2 Handling of Outliers

We adopt the strategy of [10] to detect outliers and reduce their negative effect on learning, by using the initial GMM models. We compute the accumulated

hinge loss incurred by violations of the large margin constraints in (6):

$$h_n = \sum_{c \neq y_n} \left[1 + \frac{1}{T_n} \sum_{t=1}^{T_n} \left(d(o_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}} + log \sum_{m \in S_{nt}} e^{(-d(o_{nt}, \mu_{cm}) - \theta_m)} \right) \right]_+ . \tag{8}$$

h_n measures the decrease in the loss function when an initially misclassified segment is corrected during the course of learning. We associate outliers with large values of h_n. We then re-weight the hinge loss terms by using the segment weights $s_n = min(1, 1/h_n)$:

$$\text{Ł} = \sum_{n=1}^{N} s_n h_n. \tag{9}$$

We solve this unconstrained non-linear optimization problem using the second order optimizer LBFGS [12].

4 The GSL-NAP System

In this section we briefly describe the GMM supervector linear kernel SVM system (GSL) [4] and its associated channel compensation technique, the Nuisance attribute projection (NAP) [9].

Given an M-components GMM adapted by MAP from the UBM, one forms a GMM supervector by stacking the D-dimensional mean vectors. This GMM supervector (an MD vector) can be seen as a mapping of variable-length utterances into a fixed-length high-dimensional vector, through GMM modeling:

$$\phi(\mathrm{x}) = \left[\mu_{x1} \ \cdots \ \mu_{xM} \right]^T, \tag{10}$$

where the GMM $\{\mu_{xm}, \Sigma_m, w_m\}$ is trained on the utterance x. For two utterances x and y, a kernel distance based on the Kullback-Leibler divergence between the GMM models trained on these utterances [4], is defined as:

$$\mathrm{K}(\mathrm{x}, \mathrm{y}) = \sum_{m=1}^{M} \left(\sqrt{w_m} \Sigma_m^{-(1/2)} \mu_{xm} \right)^T \left(\sqrt{w_m} \Sigma_m^{-(1/2)} \mu_{ym} \right). \tag{11}$$

The UBM weight and variance parameters are used to normalize the Gaussian means before feeding them into a linear kernel SVM training. This system is referred to as GSL in the rest of the paper.

NAP is a pre-processing method that aims to compensate the supervectors by removing the directions of undesired sessions variability, before the SVM training [9]. NAP transforms a supervector ϕ to a compensated supervector $\hat{\phi}$:

$$\hat{\phi} = \phi \ - \ \mathbf{S}(\mathbf{S}^T \phi), \tag{12}$$

using the eigenchannel matrix $\mathbf{S}$, which is trained using several recordings (sessions) of various speakers. Given a set of expanded recordings of N different

speakers, with h_i different sessions for each speaker s_i, one first removes the speakers variability by subtracting the mean of the supervectors within each speaker. The resulting supervectors are then pooled into a single matrix $\mathbf{C}$ representing the intersession variations. One identifies finally the subspace of dimension R where the variations are the largest by solving the eigenvalue problem on the covariance matrix $\mathbf{CC}^T$, getting thus the projection matrix $\mathbf{S}$ of a size $MD \times R$. This system is referred to as GSL-NAP in the rest of the paper.

5 Symmetrical Factor Analysis (SFA)

In this section we describe the symmetrical variant of the Factor Analysis model (SFA) [8] (Factor Analysis was originally proposed in [6, 7]).

In the mean supervector space, a speaker model can be decomposed into three different components: a session-speaker independent component (the UBM model), a speaker dependent component and a session dependent component. The session-speaker model, can be written as [8]:

$$\mathbf{M}_{(h,s)} = \mathbf{M} + \mathbf{D}\mathbf{y}_s + \mathbf{U}\mathbf{x}_{(h,s)}, \tag{13}$$

where

- $\mathbf{M}_{(h,s)}$ is the session-speaker dependent supervector mean (an MD vector),
- $\mathbf{M}$ is the UBM supervector mean (an MD vector),
- $\mathbf{D}$ is a $MD \times MD$ diagonal matrix, where $\mathbf{DD}^T$ represents the a priori covariance matrix of $\mathbf{y}_s$,
- $\mathbf{y}_s$ is the speaker vector, i.e., the speaker offset (an MD vector),
- $\mathbf{U}$ is the session variability matrix of low rank R (an $MD \times R$ matrix),
- $\mathbf{x}_{(h,s)}$ are the channel factors, i.e., the session offset (an R vector not dependent on s in theory).

$\mathbf{D}\mathbf{y}_s$ and $\mathbf{U}\mathbf{x}_{(h,s)}$ represent respectively the speaker dependent component and the session dependent component.

The factor analysis modeling starts by estimating the $\mathbf{U}$ matrix, using different recordings per speaker. Given the fixed parameters $(\mathbf{M}, \mathbf{D}, \mathbf{U})$, the target models are then compensated by eliminating the session mismatch directly in the model domain. Whereas, the compensation in the test is performed at the frame level (feature domain).

6 Experimental Results

We perform experiments on the NIST-SRE'2006 speaker identification task and compare the performances of the baseline GMM, the LM-dGMM and the SVM systems, with and without using channel compensation techniques. The comparisons are made on the male part of the NIST-SRE'2006 core condition (1conv4w-1conv4w). The feature extraction is carried out by the filter-bank based cepstral analysis tool Spro [13]. Bandwidth is limited to the 300-3400Hz range. 24 filter

Table 1. Speaker identification rates with GMM, Large Margin diagonal GMM and GSL models, with and without channel compensation.

System	256 Gaussians	512 Gaussians
GMM	76.46%	77.49%
LM-dGMM	77.62%	78.40%
GSL	81.18%	82.21%
LM-dGMM-SFA	89.65%	**91.27%**
GSL-NAP	87.19%	87.77%

bank coefficients are first computed over 20ms Hamming windowed frames at a 10ms frame rate and transformed into Linear Frequency Cepstral Coefficients (LFCC). Consequently, the feature vector is composed of 50 coefficients including 19 LFCC, their first derivatives, their 11 first second derivatives and the delta-energy. The LFCCs are preprocessed by Cepstral Mean Subtraction and variance normalization. We applied an energy-based voice activity detection to remove silence frames, hence keeping only the most informative frames. Finally, the remaining parameter vectors are normalized to fit a zero mean and unit variance distribution.

We use the state-of-the-art open source software ALIZE/Spkdet [14] for GMM, SFA, GSL and GSL-NAP modeling. A male-dependent UBM is trained using all the telephone data from the NIST-SRE'2004. Then we train a MAP adapted GMM for the 349 target speakers belonging to the primary task. The corresponding list of 539554 trials (involving 1546 test segments) are used for test. Score normalization techniques are not used in our experiments. The so MAP adapted GMM define the baseline GMM system, and are used as initialization for the LM-dGMM one. The GSL system uses a list of 200 impostor speakers from the NIST-SRE'2004, on the SVM training. The LM-dGMM-SFA system is initialized by model domain compensated GMM, which are then discriminated using feature domain compensated data. The session variability matrix $\mathbf{U}$ of SFA and the channel matrix $\mathbf{S}$ of NAP, both of rank $R = 40$, are estimated on NIST-SRE'2004 data using 2934 utterances of 124 different male speakers.

Table 1 shows the speaker identification accuracy scores of the various systems, for models with 256 and 512 Gaussian components ($M = 256, 512$). All these scores are obtained with the 10 best proxy labels selected using the UBM, $k = 10$. The results of Table 1 show that, without SFA channel compensation, the LM-dGMM system outperforms the classical generative GMM one, however it does yield worse performances than the discriminative approach GSL. Nonetheless, when applying channel compensation techniques, GSL-NAP outperforms GSL as expected, but the LM-dGMM-SFA system significantly outperforms the GSL-NAP one. Our best system achieves 91.27% speaker identification rate, while the best GSL-NAP achieves 87.77%. This leads to a 3.5% improvement. These results show that our fast Large Margin GMM discriminative learning algorithm not only allows efficient training but also achieves better speaker identification accuracy than a state-of-the-art discriminative technique.

7 Conclusion

We presented a new fast algorithm for discriminative training of Large-Margin diagonal GMM by using the k-best scoring Gaussians selected form the UBM. This algorithm is highly efficient which makes it well suited to process large scale databases. We carried out experiments on a full speaker identification task under the NIST-SRE'2006 core condition. Combined with the SFA channel compensation technique, the resulting algorithm significantly outperforms the state-of-the-art speaker recognition discriminative approach GSL-NAP. Another major advantage of our method is that it outputs diagonal GMM models. Thus, broadly used GMM techniques/softwares such as SFA or ALIZE/Spkdet can be readily applied in our framework. Our future work will consist in improving margin selection and outliers handling. This should indeed improve the performances.

References

1. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker Verification Using Adapted Gaussian Mixture Models. Digit. Signal Processing 10, 1-3, 19–41 (2000)
2. Keshet, J., Bengio, S.: Automatic Speech and Speaker Recognition: Large Margin and Kernel Methods. Wiley (2009)
3. Louradour, J., Daoudi, K., Bach, F.: Feature Space Mahalanobis Sequence Kernels: Application to Svm Speaker Verification. IEEE Trans. Audio Speech Lang. Processing 15, 8, 2465–2475 (2007)
4. Campbell, W.M., Sturim, D.E., Reynolds, D.A.: Support Vector Machines Using GMM Supervectors for Speaker Verification. IEEE Signal Processing Lett. 13, 5, 308–311 (2006)
5. Campbell, W.M., Sturim, D.E., Reynolds, D.A., Solomonoff, A.: SVM Based Speaker Verification Using a GMM Supervector Kernel and NAP Variability Compensation. In: ICASSP, vol. 1, pp. I-97–I-100 (2006)
6. Kenny, P., Boulianne, G., Dumouchel, P.: Eigenvoice Modeling with Sparse Training Data. IEEE Trans. Speech Audio Processing 13, 3, 345–354 (2005)
7. Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P.: Speaker and Session Variability in GMM-Based Speaker Verification. IEEE Trans. Audio Speech Lang. Processing 15, 4, 1448–1460 (2007)
8. Matrouf, D., Scheffer, N., Fauve, B.G.B., Bonastre, J.-F.: A Straightforward and Efficient Implementation of the Factor Analysis Model for Speaker Verification. In: Interspeech, pp. 1242–1245 (2007)
9. Solomonoff, A., Campbell, W.M., Quillen, C.: Nuisance Attribute Projection. Speech Communication (2007)
10. Sha, F., Saul, L.K.: Large Margin Gaussian Mixture Modeling for Phonetic Classification and Recognition. In: ICASSP, vol. 1, pp. 265–268 (2006)
11. Jourani, R., Daoudi, K., André-Obrecht, R., Aboutajdine, D.: Large Margin Gaussian Mixture Models for Speaker Identification. In: Interspeech, pp. 1441–1444 (2010)
12. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer verlag (1999)
13. Gravier, G.: SPro: "Speech Signal Processing Toolkit", `https://gforge.inria.fr/projects/spro` (2003)
14. Bonastre, J.-F. et al.: ALIZE/SpkDet: a State-of-the-art Open Source Software for Speaker Recognition. In: Odyssey (2008)