Speaker verification using Large Margin GMM discriminative training

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

Speaker verification using Large Margin GMM discriminative training

Reda Jourani * ‡, Khalid Daoudi †, Régine André-Obrecht * and Driss Aboutajdine ‡
* SAMoVA Group, IRIT - UMR 5505 du CNRS
University Paul Sabatier, 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France
† INRIA Bordeaux-Sud Ouest
351, cours de la libération. 33405 Talence. France
‡ Laboratoire LRIT. Faculty of Sciences, Mohammed 5 Agdal University
4 Av. Ibn Battouta B.P. 1014 RP, Rabat, Morocco
{jourani, obrecht}@irit.fr, khalid.daoudi@inria.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. The resulting algorithm is thus well suited to handle large scale databases. To show the effectiveness of the new algorithm, we carry out a full NIST speaker verification task using NIST-SRE'2006 data. The results show that our system outperforms the baseline GMM, and with high computational efficiency.**

***Index Terms*—Large margin training, Gaussian mixture models, discriminative learning, speaker recognition, speaker verification**

I. 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 [1]. Generative training does not however directly optimize the classification performance since it provides a model for the joint probability distribution. For this reason, discriminative training approaches have been an interesting and valuable alternative since they address directly the classification problem [2], and lead generally 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 [3].

Recently a new discriminative approach for multiway classification has been proposed, the Large Margin Gaussian mixture models (LM-GMM) [4]. 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, and thus no kernel trick/matrix is required. 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 [5], we proposed a simplified version of LM-GMM which exploit the fact that traditional GMM 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 scale databases such as in NIST Speaker Recognition Evaluation (NIST-SRE) campaigns.

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. We also address a speaker verification task which is a more difficult task than speaker identification. 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 UBM mixtures. In order to show the effectiveness of the new algorithm, we carry out a full NIST speaker verification task using NIST-SRE'2006 (core condition) data. The results show that our new algorithm is not only highly efficient but also outperforms the baseline generative GMM.

The paper is organized as follows. After an overview on Large-Margin GMM training in section 2, we describe our new training algorithm in section 3. Experimental results are then reported in section 4.

II. Overview on Large Margin GMM training

In this section we start by recalling the original Large Margin GMM training algorithm developed in [4], [6]. We then recall the simplified version of this algorithm that we introduced in [5].

A. Large Margin GMM

In Large Margin GMM [4], [6], 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 parametrized by a centroid vector μ_{cm} (mean vector), 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 $\{x_{nt}\}_{t=1}^{T_n}$ ($x_{nt} \in \mathcal{R}^D$) be the T_n feature vectors of the n^{th} segment (i.e. n^{th} speaker training data). Then, for each x_{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:

$$\forall c \neq y_n, \quad -log \sum_{m=1}^{M} e^{-z_{nt}^T \Phi_{cm} z_{nt}} \geq 1 + z_{nt}^T \Phi_{y_n m_{nt}} z_{nt}, \quad (2)$$

where $z_{nt} = \begin{bmatrix} x_{nt} \\ 1 \end{bmatrix}$.

Because of the softmax inequality: $min_m\ a_m \geq -log \sum_m e^{-a_m}$, Eq. (2) states that for each competing class $c \neq y_n$ the match (in term of Mahalanobis distance) of any centroid in class c is worse than the target centroid by a margin of at least one unit.

In a segmental training scheme, the loss function is thus given by:

$$Ł = \sum_{n=1}^{N} \sum_{c \neq y_n} max\left(0\ ,\ 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} \left(z_{nt}^T \Phi_{y_n m_{nt}} z_{nt} + log \sum_{m=1}^{M} e^{-z_{nt}^T \Phi_{cm} z_{nt}} \right)\right) + \alpha \sum_{cm} trace(\Psi_{cm}), \quad (3)$$

where the second term penalizes large trace Mahalanobis metrics. The hyperparameter α is set by cross-validation on development data.

Finally, the decision rule used for classification is:

$$y = argmin_c \left\{ \sum_{t=1}^{T} -log \sum_{m=1}^{M} e^{-z_t^T \Phi_{cm} z_t} \right\}. \quad (4)$$

As opposed to other discriminative training algorithms such as conditional log-likelihood learning, the major advantage of this loss function is its convexity. For a complete description of the LM-GMM and their extension to LM-HMM, we refer to [4], [6], [7].

B. Large Margin GMM with diagonal covariances (LM-dGMM)

Most of state-of-the art speaker recognition 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 [8] from tens or hundreds of hours of speech data gathered from a large number of speakers. The background model represents speaker-independent distribution of the feature vectors. When enrolling a new speaker to the system, the parameters of the UBM are adapted to the feature distribution of the new speaker. The adapted model is then used as the model of that 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], while the (diagonal) covariances and the weights remain unchanged.

Following the same philosophy of traditional GMM, we proposed in [5] to neglect the orientation of the Ψ_{cm} matrices in training. That is, in our Large Margin diagonal GMM (LM-dGMM) [5], 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 parametrized 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.

With this relaxation on the matrices Ψ_{cm}, for each example x_{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 $\{(x_{nt}, y_n, m_{nt})\}_{n=1}^{N}$, we seek mean vectors μ_{cm} which satisfy the LM-dGMM criterion:

$$\begin{aligned} &\forall c \neq y_n,\ \forall m, \\ &d(x_{nt}, \mu_{cm}) + \theta_m \geq 1 + d(x_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}}, \end{aligned} \quad (5)$$

where $d(x_{nt}, \mu_{cm}) = \sum_{i=1}^{D} \frac{(x_{nti} - \mu_{cmi})^2}{2\sigma_{mi}^2}$.

Afterward, these M constraints are fold into a single one using the softmax inequality. The segment-based LM-dGMM criterion becomes thus:

$$\begin{aligned} &\forall c \neq y_n, \\ &\tfrac{1}{T_n} \sum_{t=1}^{T_n} \left(-log \sum_{m=1}^{M} exp(-d(x_{nt}, \mu_{cm}) - \theta_m) \right) \\ &\geq 1 + \tfrac{1}{T_n} \sum_{t=1}^{T_n} d(x_{nt}, \mu_{y_n m_{nt}} + \theta_{m_{nt}}). \end{aligned} \quad (6)$$

The loss function to minimize for LM-dGMM is then given by:

$$Ł = \sum_{n=1}^{N} \sum_{c \neq y_n} max\left(0\ ,\ 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} \left(d(x_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}} + log \sum_{m=1}^{M} exp(-d(x_{nt}, \mu_{cm}) - \theta_m) \right)\right). \quad (7)$$

As compared to the original algorithm, we showed in [5] that this simplified version has the advantage of being more efficient while it still yields similar or better performances on a speaker identification task.

III. LM-dGMM TRAINING WITH k-BEST GAUSSIANS

A. 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 [4], [6], we still encountered efficiency problems when dealing with high number of Gaussian mixtures. Indeed, even for an easy 50 speakers identification task as the one presented in [5], we could not run the training in a relatively short time with our current implementation. This would imply that large scale applications such as NIST-SRE, where hundreds or thousands of target speakers are available, would be infeasible in reasonable time (for instance, 5460 target speakers are included in the NIST-SRE'2010 core condition, with 610748 trials to process involving 13325 test segments [9]).

In order to develop a fast training algorithm which could be used in large scale applications, we propose to drastically reduce the number of constraints to satisfy in Eq. (6). By doing so, we would drastically reduce the computational complexity of the loss function and its gradient, which are the quantities responsible for most of the computational time. 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 x_n and each c, instead of summing over the M mixtures in the left side of equation Eq. (6), 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 x_{nt}, instead of $(C-1)$ sets. This leads to a $(C-1)$ times faster and less memory consuming selection. Thus, the higher the number of target speakers is, the greater computation and memory saving is.

More precisely, we now seek mean vectors μ_{cm} that satisfy the large margin constraints in Eq. (8) :

$$\begin{aligned} &\forall c \neq y_n, \\ &\tfrac{1}{T_n}\sum_{t=1}^{T_n}\left(-\,log \sum_{m\in S_{nt}} exp(-d(x_{nt},\mu_{cm})-\theta_m)\right) \\ &\geq 1+\tfrac{1}{T_n}\sum_{t=1}^{T_n} d(x_{nt},\mu_{y_n m_{nt}}+\theta_{m_{nt}}). \end{aligned} \tag{8}$$

The loss function becomes:

$$\begin{aligned} Ł \;=\; \sum_{n=1}^{N}\sum_{c\neq y_n} max\Bigg(0\;,\; 1+\frac{1}{T_n}\sum_{t=1}^{T_n}\Bigg(d(x_{nt},\mu_{y_n m_{nt}}) \\ +\,\theta_{m_{nt}}+log\sum_{m\in S_{nt}} exp(-d(x_{nt},\mu_{cm})-\theta_m)\Bigg)\Bigg). \end{aligned} \tag{9}$$

This loss function remains convex and can still be solved using dynamic programming.

During test, we compute a match score depending on both the target model $\{\mu_{cm},\Sigma_m,\theta_m\}$ and the UBM $\{\mu_{Um},\Sigma_m,\theta_m\}$ for each test hypothesis. 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 average log likelihood ratio using only these k labels:

$$\begin{aligned} LLR_{avg} \;=\; \tfrac{1}{T}\sum_{t=1}^{T}\Bigg(log\sum_{m\in E_t}\exp(-d(x_t,\mu_{cm})-\theta_m) \\ -\,log\sum_{m\in E_t}\exp(-d(x_t,\mu_{Um})-\theta_m)\Bigg). \end{aligned} \tag{10}$$

This quantity provides a score for the the test segment to be uttered by the target model/speaker c. The higher the score is, the greater the probability that the test segment was uttered by the target speaker is.

B. Handling of outliers

We adopt the strategy of [4] to detect outliers and reduce their negative effect on learning. Outliers are detected using the initial GMM models. We compute the accumulated hinge loss incurred by violations of the large margin constraints in Eq. (8) :

$$\begin{aligned} h_n \;=\; \sum_{c\neq y_n} max\Bigg(0\;,\; 1+\frac{1}{T_n}\sum_{t=1}^{T_n}\Bigg(d(x_{nt},\mu_{y_n m_{nt}}) \\ +\,\theta_{m_{nt}}+log\sum_{m\in S_{nt}} exp(-d(x_{nt},\mu_{cm})-\theta_m)\Bigg)\Bigg). \end{aligned} \tag{11}$$

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 in Eq. (9) by using segment weights $sw_n = min(1,\frac{1}{h_n})$:

$$Ł \;=\; \sum_{n=1}^{N} sw_n h_n. \tag{12}$$

We solve this unconstrained non-linear optimization problem using the second order optimizer LBFGS [10].

In summary, our new and fast training algorithm of LM-dGMM is the following:

- For each class (speaker), initialize with the GMM trained by MAP of the UBM,
- select Proxy labels using these GMM,
- select the set of k-best UBM Gaussian components for each training frame,
- compute the segment weights,
- using the LBFGS algorithm, solve the unconstrained non-linear optimization problem according to equation Eq. (12)

$$min \quad Ł. \tag{13}$$

TABLE I
EER(%) AND minDCF(x100) performances for GMM and LM-dGMM systems with and without T-norm, using models with 256 Gaussian components.

System	no T-norm		with T-norm	
	EER	minDCF	EER	minDCF
GMM	9.48	4.26	8.83	3.56
LM-dGMM	8.97	3.97	**8.40**	3.49

IV. Experimental results

We perform experiments on the NIST-SRE'2006 [11] speaker verification task and compare the performances of the baseline GMM and our new LM-dGMM system. The comparisons are made on the male part of the NIST-SRE'2006 core condition (1conv4w-1conv4w). Performances are assessed using Detection Error Tradeoff (DET) plots and measured in terms of equal error rate (EER) and minimum of detection cost function (minDCF). The latter is calculated following NIST criteria [12].

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 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], [15] for GMM 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 22123 trials (involving 1601 test segments) are used for test. T-norm score normalization technique [16] is applied to the log-likelihood ratio scores. Session variability modeling techniques are not used in our experiments. 200 male speakers from NIST-SRE'2004 are used as background data. The so MAP adapted GMM define the baseline GMM system, and are used as initialization for the LM-dGMM one.

Table I provides the EERs and minDCFs of the two systems, with and without T-norm, for models with 256 Gaussian components ($M = 256$). Figure 1 shows DET plots for the best GMM and LM-dGMM systems (with T-norm). All these results are obtained with the 10 best proxy labels selected using the UBM, $k = 10$.

The results show that the LM-dGMM algorithm yields better performances than the GMM system. In particular, our best system achieves 8.40% equal error rate, while the best GMM achieves 8.83%. This leads to a relative reduction

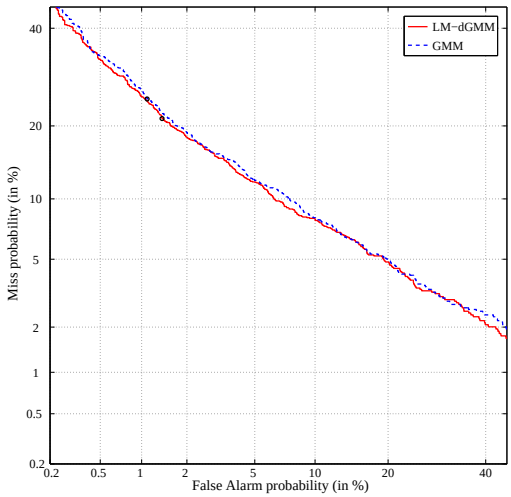

Fig. 1. *DET plots for GMM and LM-dGMM systems with T-normalization.*

of EER of about 4.87%. These results suggest that our k-best technique not only allow efficient training but also still outperforms the baseline generative GMM system. We mention here that we observed the same behavior of our new algorithm on the speaker identification task presented in [5]. We can thus fairly consider that our fast Large Margin GMM discriminative training algorithm is a good alternative to the classical generative GMM training in the setting of speaker recognition. We also expect further performance improvements when combining it with other discriminative methods such SVM-GMM supervectors [3].

V. Conclusion

We presented a new simplified algorithm to train Large-Margin 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 such as in NIST'SRE. We carried out experiments on a speaker verification task under the NIST-SRE'2006 core condition. The results show that we achieve better accuracy than the baseline GMM system (trained with ALIZE/Spkdet) with high computational efficiency. These results suggest that this framework is promising should be further investigated and compared/combined with other discriminative methods, such as SVM-GMM supervectors in particular. This will be the purpose of future communications. We also emphasize that while we have been interested in speaker recognition applications, our algorithm can be used in many other classification applications involving large training databases.

References

[1] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[2] J. Keshet and S. Bengio, *Automatic speech and speaker recognition: Large margin and kernel methods.* Wiley, 2009.

[3] W. Campbell, D. Sturim, and D. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, 2006.

[4] F. Sha and L. Saul, "Large margin Gaussian mixture modeling for phonetic classification and recognition," in *Proc. of ICASSP*, vol. 1. IEEE, 2006, pp. 265–268.

[5] R. Jourani, K. Daoudi, R. André-Obrecht, and D. Aboutajdine, "Large Margin Gaussian mixture models for speaker identification," in *Proc. of Interspeech*, 2010, pp. 1441–1444.

[6] F. Sha, "Large margin training of acoustic models for speech recognition," Ph.D. dissertation, University of Pennsylvania, 2007.

[7] F. Sha and L. Saul, "Large margin hidden Markov models for automatic speech recognition," *Advances in Neural Information Processing Systems*, vol. 19, pp. 1249–1256, 2007.

[8] C. Bishop, *Pattern recognition and machine learning*. Springer Science+Business Media, LLC, New York, 2006.

[9] *The NIST Year 2010 Speaker Recognition Evaluation Plan*, 2010, online: http://www.itl.nist.gov/iad/mig//tests/sre/2010/index.html.

[10] J. Nocedal and S. Wright, *Numerical optimization*. Springer verlag, 1999.

[11] *The NIST Year 2006 Speaker Recognition Evaluation Plan*, 2006, online: http://www.itl.nist.gov/iad/mig/tests/spk/2006/index.html.

[12] M. Przybocki and A. Martin, "NIST Speaker Recognition Evaluation Chronicles," in *Proc. of Odyssey-The Speaker and Language Recognition Workshop*, 2004, pp. 15–22.

[13] G. Gravier, *SPro: "Speech Signal Processing Toolkit"*, 2003, online: http://www.gforge.inria.fr/projects/spro.

[14] J.-F. Bonastre, N. Scheffer, D. Matrouf, C. Fredouille, A. Larcher, A. Preti, G. Pouchoulin, N. Evans, B. Fauve, and J. Mason, "ALIZE/SpkDet: a state-of-the-art open source software for speaker recognition," in *Proc. of Odyssey-The Speaker and Language Recognition Workshop*, 2008.

[15] B. Fauve, D. Matrouf, N. Scheffer, J.-F. Bonastre, and J. Mason, "State-of-the-Art Performance in Text-Independent Speaker Verification through Open-Source Software," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, Issue 7, pp. 1960–1968, 2007.

[16] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score Normalization for Text-Independent Speaker Verification Systems," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 42–54, 2000.