Large Margin Gaussian mixture models for speaker identification

Reda Jourani\textsuperscript{1,3}, Khalid Daoudi\textsuperscript{2}, Régine André-Obrecht\textsuperscript{1} and Driss Aboutajdine\textsuperscript{3}

\textsuperscript{1}SAMoVA Group, IRIT - UMR 5505 du CNRS
University Paul Sabatier, 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France
\textsuperscript{2}INRIA Bordeaux-Sud Ouest
351, cours de la libération. 33405 Talence. France
\textsuperscript{3}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 decade. However, they are generally trained using the generative criterion of maximum likelihood estimation. In this paper, we propose a simple and efficient discriminative approach to learn GMM with a large margin criterion to solve the classification problem. Our approach is based on a recent work about the Large Margin GMM (LM-GMM) where each class is modeled by a mixture of ellipsoids and which has shown good results in speech recognition. We propose a simplification of the original algorithm and carry out preliminary experiments on a speaker identification task using NIST-SRE’2006 data. We compare the traditional generative GMM approach, the original LM-GMM one and our own version. The results suggest that our algorithm outperforms the two others.

Index Terms: large margin learning, GMM-UBM, speaker recognition, discriminative and generative learning.

1. Introduction

The closed set speaker identification task consists of determining, from a sequence of speech samples, the identity of an unknown person among N reference speakers. Most of the speaker identification systems rely on the generative approach with the classical Gaussian Mixture Models (GMM) [1]; they have the advantage to be text independent and insensitive to the temporal variability of speech, but their training by maximum likelihood estimation (MLE) or maximum a posteriori (MAP) does not directly optimize the classification performance. Discriminative approaches have been an interesting and valuable alternative to address directly the classification problem [2]. For instance, Support Vector Machines (SVM) combined with GMM supervectors are among state-of-the-art approaches in speaker recognition [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 is required. This way no large kernel matrix has to be maintained, leading to potentially more tractable training in large scale applications. While LM-GMM (and their LM-HMM extension) have been used in speech recognition, they have not been used in speaker recognition (to the best of our knowledge). In this paper, we propose a simplified version of LM-GMM which has the major advantage to lead to a very efficient training algorithm for speaker recognition. We do so by following the same philosophy of traditional GMM where the MAP adaptation of target speaker models is done only on the mean vectors. We perform preliminary experiments on NIST-SRE data. The results suggest that our simplified algorithm outperforms both the original LM-GMM and the traditional GMM. After a brief overview on Large Margin GMM in section 2, we describe our simplified LM-GMM training algorithm in section 3. In section 4, are reported experimental results on speaker identification.

2. A brief overview on Large Margin GMM (LM-GMM)

In Large Margin GMM, each class is modeled by a mixture of ellipsoids in the $D$-dimensional input space. For each class $c$, the $m^{th}$ ellipsoid is parametrized by a centroid vector $\mu_{cm}$ (mean vector), a positive semidefinite (orientation) matrix $\Psi_{cm}$ and a nonnegative scalar offset $\theta_{cm} \geq 0$.

For each $c$ and $m$, the parameters $\mu_{cm}$, $\Psi_{cm}$, $\theta_{cm}$ are then collected into a single enlarged matrix $\Phi_{cm}$:

$$\Phi_{cm} = \begin{pmatrix} \Psi_{cm} & \mu_{cm}^T \\ \mu_{cm} & \Psi_{cm}^T \mu_{cm} + \theta_{cm} \end{pmatrix}.$$ (1)
Considering a set of labeled training examples \( \{(x_n, y_n)\}_{n=1}^{N} \) where \( x_n \in \mathbb{R}^L \) and \( y_n \in \{1, 2, \ldots, C\} \), the goal of LM-GMM training is to find matrices \( \Phi_{cm} \) such that "all" examples are correctly classified by at least one margin unit. To do so, a GMM is first fit to each class using maximum likelihood estimation. Second, a label (proxy label) \( m_n \) is associated to each example \( x_n \); this label corresponds to the GMM mixture component with the highest posterior probability.

Given the joint labels \( (y_n,m_n) \) for each learning example, the LM-GMM criterion is given as:

\[
\forall c \neq y_n, \quad -\log \sum_m e^{-z_n^T \Phi_{cm} z_n} - z_n^T \Phi_{y_n,m_n} z_n \geq 1,
\]  

(2)

where \( z_n = \begin{bmatrix} x_n \\ 1 \end{bmatrix} \).

Because of the softmax inequality:

\[ m_n \geq -\log \sum_m e^{-y_n m_n}, \quad \text{Eq. (2) states that} \]

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.

The loss function for LM-GMM is thus given by:

\[
L = \sum_n \sum_{c \neq y_n} \max \left( 0, 1 + z_n^T \Phi_{y_n,m_n} z_n + \log \sum_m e^{-z_n^T \Phi_{cm} z_n} \right) + \alpha \sum_{cm} \text{trace}(\Psi_{cm}),
\]

(3)

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

The decision rule used for classification is:

\[
y = \arg\min_c \left\{ -\log \sum_m \exp \left( -z_n^T \Phi_{cm} z_n \right) \right\}.
\]

(4)

As opposed to other discriminative training algorithms such as conditional log-likelihood learning, the major advantage of this loss function is that it is convex. For a complete description of the LM-GMM and their extension to LM-HMM, we refer to [4, 5, 6].

3. LM-GMM for speaker recognition

The majority of state-of-the-art speaker recognition systems, particularly in NIST-SRE campaigns, 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 [7] 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.

3.1. Simplified LM-GMM

Following the same philosophy of traditional GMM, it is natural to think about neglecting the orientation of the \( \Psi_{cm} \) matrices. With this purpose in mind, we now assume that each class \( c \) is modeled by a mixture of ellipsoids with parameters \( \{\mu_{cm}, \Psi_{cm}, \theta_m\} \), where each \( \Psi_m = \text{diag}(\frac{1}{\sigma_{cm,1}^2}, \ldots, \frac{1}{\sigma_{cm,d}^2}) \) is diagonal. We then seek only the centroid vectors \( \mu_{cm} \) which are derived from the UBM model by updating only the \( \mu_m \) and \( \sigma_m \) in the GMM-UBM approach, the target speaker GMM is adapted to the feature distribution of the new speaker.

\[
y = \arg\min_c \left\{ -\log \sum_m \exp \left( -z_n^T \Phi_{cm} z_n \right) \right\}.
\]

(7)

Besides its philosophy, which seems appropriate to speaker recognition, this simplified algorithm has a major advantage as compared to the original one: computational efficiency. Indeed, with this simplified version the matrices \( \Phi_{cm} \) disappear and there is no need to guarantee semidefinite positiveness (by square-rooting as done in [5]) during optimization. This results in a very low complexity algorithm which can be suited for large scale applications, such as NIST-SRE campaigns.

3.2. Segmental training

As in the original algorithm, we rewrite the previous frame-based formulas in the segmental training scheme,
to apply collectively to multiple consecutive analysis frames. Let \( t \) index the \( T_n \) frames belonging to the \( n^{th} \) segment (i.e. \( n^{th} \) speaker training data) \( \{x_{nt} \}_{t=1}^{T_n} \).

The segment-based large margin constraints, loss function and decision rule are thus:

\[
\begin{align*}
\forall c \neq y_n, \\
\frac{1}{T_n} \sum_t \left( - \log \sum_m \exp (-d(x_{nt}, \mu_{cm}) - \theta_m) \right) \\
\geq 1 + \frac{1}{T_n} \sum_t d(x_{nt}, \mu_{y_n,m_n} + \theta_{m_n}).
\end{align*}
\]  
(8)

\[
L = \sum_n \max_{c \neq y_n} \left( 0, 1 + \frac{1}{T_n} \sum_t \left( d(x_{nt}, \mu_{y_n,m_n}) \\
+ \theta_{m_n} + \log \sum_m \exp (-d(x_{nt}, \mu_{cm}) - \theta_m) \right) \right) + \alpha \sum_{md} \frac{1}{\sigma_{md}^2},
\]  
(9)

\[
y = \arg \min_{c} \left\{ \sum_t - \log \sum_m \exp (-d(x_t, \mu_{cm}) - \theta_m) \right\}.
\]  
(10)

3.3. 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):

\[
h_n = \sum_{c \neq y_n} \max \left( 0, 1 + \frac{1}{T_n} \sum_t \left( d(x_{nt}, \mu_{y_n,m_n}) + \theta_{m_n} + \log \sum_m \exp (-d(x_{nt}, \mu_{cm}) - \theta_m) \right) \right).
\]  
(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 \( h_n \). And we re-weight the hinge loss terms in Eq. (9) by using segments weights \( sw_n = \min(1, \frac{1}{h_n}) \):

\[
L = \sum_n sw_n h_n + \alpha \sum_{md} \frac{1}{\sigma_{md}^2}.
\]  
(12)

We solve this unconstrained non-linear optimization problem using the second order optimizer LBFGS [8].

In summary, our simplified LM-GMM training algorithm is the following:

- For each class (speaker), initialize with the GMM trained by MAP of the UBM,
- select Proxy labels using these GMM,
- compute the segments weights,
- Using the LBFGS algorithm, solve the unconstrained non-linear optimization problem according to equation Eq. (12)

\[
min \ L.
\]  
(13)

4. Experimental results

We compare the three systems: the traditional GMM system, the original LM-GMM one and our modified version. We perform experiments on NIST-SRE’2004 and 2006 data and evaluate performance for the task of speaker identification. The feature extraction is carried out by the filter-bank based cepstral analysis tool Spro [9]. 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 [10] for GMM modeling. The code for the original LM-GMM modeling was kindly given to us by Fei Sha.

A male-dependent UBM is trained using all the telephone data from the NIST-SRE’2004. Then we train a MAP adapted GMM for each speaker of 50 male target speakers belonged to the NIST-SRE’2006 primary task (1conv4w-1conv4w). A corresponding list of 11600 trials are used for testing. Session variability modeling and score normalization techniques are not used in our experiments. The so MAP adapted GMM are used to define the traditional GMM system. They are used as initialization for the two LM-GMM systems (original and simplified).

Table 1 shows the speaker identification accuracy scores of the three systems. For each one, we study two configurations with 16 and 32 Gaussian components.

Before discussing the results, we underline the fact that we have used no development data for the LM-GMM systems, we thus set \( \alpha = 0 \).

The results of Table 1 show that our simplified LM-GMM algorithm yields significantly better scores than the GMM system and the improvement is higher when less Gaussians are used. This is consistent with the well known fact that discriminative models perform better when the "true" distribution is not correctly captured.
Table 1: Speaker identification rates with GMM and the original and simplified Large Margin Training algorithms.

<table>
<thead>
<tr>
<th>System</th>
<th>16 Gaussians</th>
<th>32 Gaussians</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMM</td>
<td>61.6%</td>
<td>68.1%</td>
</tr>
<tr>
<td>LM-GMM</td>
<td>60.8%</td>
<td>72.0%</td>
</tr>
<tr>
<td>Simplified LM-GMM</td>
<td>67.7%</td>
<td>71.6%</td>
</tr>
</tbody>
</table>

by the generative model. One can thus fairly expect that behavior of our algorithm should remain the same in large scale and benchmark evaluations. This behavior could even improve if development data are available.

The results of Table 1 show also that our algorithm performs as well as the original one with 32 Gaussians, and significantly outperforms it with 16 Gaussians. These results suggest that the strategy of discriminating only the mean vectors is worth, and that the use the $\Phi_{cm}$ matrices could even degrade the performances while they are computationally (relatively) demanding. Indeed, computing gradients of the loss function with respect to the enlarged matrices $\Phi_{cn}$ is much more time and memory consuming than with respect to only the mean vectors $\mu_{cn}$. Moreover, our algorithm requires less iterations than the original one to converge.

Another potentially major advantage of the simplified LM-GMM is that we still have normalized GMM after the discriminative training, which is not the case with the original algorithm. This can be extremely useful for post-processing, for instance by integrating the resulting discriminative GMM into ALIZE/Spkdet.

5. Conclusion and future work

We proposed a very simple and efficient implementation of large margin GMM which is well suited for speaker recognition tasks. The results show that our approach is very promising and should be further investigated. Our next objective is to perform experiments on full NIST-SRE tasks and compare the performances to generative GMM and GMM-supervectors SVM. We will then integrate Latent Factor Analysis into our training and testing and compare to state-of-the-art NIST-SRE systems.

6. Acknowledgements

The authors would like to thank Fei Sha for providing the Large Margin Training algorithm code. The first author would like also to thank Fei Sha for his valuable supervision and his hospitality during his 2009 summer internship at USC.

7. References