Apprentissage discriminant des GMM à grande marge pour la vérification automatique du locuteur

Reda JOURANI [1 3],Khalid DAOUDI [2],Régine ANDRÉ-OBRECHT [1],Driss ABOUTAJDINE [3]

[1] Équipe SAMoVA, IRIT - UMR 5505 du CNRS
Université Paul Sabatier, 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France

[2] INRIA Bordeaux-Sud Ouest
351, cours de la libération. 33405 Talence. France

[3] Laboratoire LRIT. Faculté des Sciences de Rabat, Université Mohammed V Agdal
4 Av. Ibn Battouta B.P. 1014 RP, Rabat, Maroc

{jourani, obrecht}@irit.fr, khalid.daoudi@inria.fr, aboutaj@fsr.ac.ma

Résumé – Les modèles de mélange de Gaussiennes (GMM) sont communément utilisés en reconnaissance automatique du locuteur. Ils sont généralement appris par une approche générative basée sur les techniques de maximum de vraisemblance et de maximum a posteriori. Dans un travail précédent, nous avons proposé un algorithme d'apprentissage discriminant des GMM (à matrices de covariance diagonales) minimisant une fonction de perte à grande marge. Nous présentons dans cette communication, une nouvelle version plus rapide de cet algorithme qui permet de traiter de grands volumes de données. Des tests effectués sur la tâche de vérification du locuteur de la compagne NIST-SRE'2006 montrent que notre système donne de meilleurs résultats que les modèles GMM génératifs.

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.

1 Introduction

Tout système de vérification automatique de locuteurs essaie de répondre à la question suivante : "étant donné un locuteur cible (un modèle apprit sur des données d'apprentissage) et un segment (de parole) de test, déterminer si le locuteur cible parle dans se segment, en accompagnant la décision (Vraie ou Faux) par un score (de préférence, un rapport de log-vraisemblance)".

La majorité des systèmes actuels de reconnaissance automatique du locuteur (RAL) sont basés sur l'utilisation de modèles de mélange de Gaussiennes (GMM). Ces modèles sont généralement appris par une approche générative en utilisant les techniques de maximum de vraisemblance et de maximum a posteriori [1]. Cependant, cet apprentissage génératif ne s'attaque pas directement au problème de classification étant donné qu'il fournit un modèle à la distribution jointe. Ceci a conduit récemment à l'émergence d'approches discriminantes qui tentent de résoudre directement le problème de classification [2], et qui donnent généralement de bien meilleurs résultats. Par exemple, les machines à vecteurs de support (SVM), combinées avec les supervecteurs GMM sont parmi les techniques les plus performantes en RAL [3].

Récemment, une nouvelle approche discriminante pour la séparation multi-classes a été proposée et appliquée en reconnaissance de la parole, les GMM à grande marge (LM-GMM) [4]. Cette dernière utilise la même notion de marge que les SVM et possède les même avantages que les SVM en terme de la convexité du problème à résoudre. Mais elle diffère des SVM car elle construit une frontière non-linéaire entre les classes directement dans l'espace des données. Ainsi, l'astuce du noyau (kernel trick) et la matrice de Gram (kernel matrix) ne sont pas requis. En RAL, les systèmes GMM de l'état de l'art utilisent des matrices de covariance diagonales et sont appris par adaptation MAP (maximum a posteriori) des vecteurs moyennes d'un modèle du monde. Dans un travail précédent [5], nous avons proposé une version simplifiée des LM-GMM qui exploite cette propriété et nous l'avons appliqué à une tâche simple d'identification du locuteur. L'algorithme d'apprentissage résultant est plus simple et plus rapide que la version originale. Cependant, sa complexité algorithmique reste encore trop élevée pour traiter de grands volumes de données, tels que ceux utilisés dans les compagnes d'évaluation NIST-SRE (NIST Spea-

ker Recognition Evaluation).

Dans cette communication, nous proposons une nouvelle version de l'algorithme d'apprentissage des LM-GMM qui en plus d'être efficace a l'avantage d'être très rapide et permet ainsi de traiter de grands volumes de données. Nous appliquons ce nouvel algorithme à la tâche plus difficile (que l'identification) qu'est la vérification du locuteur. En nous plaçant dans les conditions d'évaluation de NIST-SRE'2006 [6], nous comparons ses performances aux GMM classiques et à une technique état-de-l'art en RAL largement utilisée actuellement, le *Symmetrical Factor Analysis* (SFA) [7, 8]. Le SFA consiste en un apprentissage génératif de GMM permettant de compenser la variabilité inter-sessions. Notre algorithme est basée sur deux principes : la décision de classification utilise généralement uniquement les k-meilleures gaussiennes, et la correspondance entre les composantes des GMM appris par adaptation MAP [1] et celles du modèle du monde (UBM).

Cette communication suit le plan suivant. La section 2 présente les modèles LM-GMM à matrices de covariance diagonales. Ensuite nous décrivons dans la section 3, notre nouvel algorithme qui est adapté aux grands volumes de données. Enfin, les résultats expérimentaux sont proposés dans la section 4.

2 Les modèles LM-GMM à matrices de covariance diagonales (LM-dGMM)

L'estimation par maximum de vraisemblance (l'algorithme EM [9]) donne de bons résultats quand on dispose d'une grande quantité de données, suffisamment nécessaire pour estimer robustement les paramètres d'un modèle GMM. En reconnaissance automatique du locuteurs, peu de données sont généralement disponibles pour apprendre directement ces modèles. Un modèle GMM du monde ou UBM (Universal Background Model) à matrices diagonales est ainsi appris par l'algorithme EM sur des centaines d'heures d'enregistrements appartenant à plusieurs locuteurs et dans différentes conditions. Ensuite, le modèle d'un client est appris par adaptation (généralement par la méthode de maximum a posteriori (MAP)) de l'UBM aux données de ce client. On peut adapter l'ensemble des paramètres de l'UBM, comme on peut se limiter à en adapter que certains. Reynolds a montré dans [1] que l'adaptation des moyennes uniquement donne de bons résultats. Les matrices de covariance (diagonales) et les poids restent inchangés.

Exploitant cette diagonalité, nous avons proposé dans [5] un algorithme simple pour apprendre des GMM à grande marge, les LM-dGMM. Cet algorithme initialise chaque classe (locuteur) c par un GMM à M composantes, appris par adaptation MAP. La $m^{ème}$ gaussienne est paramétrée par un vecteur de moyennes μ_{cm}, une matrice de covariance diagonale $\Sigma_m = diag(\sigma^2_{m1}, ..., \sigma^2_{mD})$ où D est la dimension des vecteurs paramétriques, et un facteur $\theta_m = \frac{1}{2}\left(Dlog(2\pi) + log|\Sigma_m|\right) - log(w_m)$ qui correspond au poids de la gaussienne.

Si $\{x_{nt}\}_{t=1}^{T_n}$ ($x_{nt} \in \mathcal{R}^D$) est la séquence des T_n vecteurs paramétriques du locuteur n, et y_n ($y_n \in \{1, 2, ..., C\}$ où C est le nombre de locuteurs) sa classe, nous déterminons pour chaque vecteur x_{nt} le label m_{nt} de la composante du $y_n^{ème}$ mélange, ayant la plus grande probabilité a posteriori. L'algorithme d'apprentissage vise à ce que pour chaque vecteur x_{nt}, sa log-vraisemblance par rapport à la composante m_{nt} soit supérieure d'au moins 1 à celles calculées par rapport à toute composante de toute autre classe. Dans l'espace des paramètres, on cherche à ce que tout vecteur x_{nt} soit plus proche de son vecteur de moyennes associé $\mu_{y_n m_{nt}}$ d'au moins une distance unitaire (marge minimale unitaire) que de tout autre vecteur de moyennes μ_{cm}. Disposant donc des données d'apprentissage $\{(x_{nt}, y_n, m_{nt})\}_{n=1}^{N}$, les contraintes LM-dGMM à satisfaire sont :

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

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

Les M précédentes contraintes sont regroupées en une seule, en utilisant l'inégalité softmax $min_m\ a_m \geq -log \sum_m e^{-a_m}$. Ainsi dans le cadre d'un apprentissage segmental, les contraintes à grande marge deviennent :

$$\begin{array}{l} \forall c \neq y_n, \\ \frac{1}{T_n} \sum_{t=1}^{T_n} \left(- log \sum_{m=1}^{M} exp(-d(x_{nt}, \mu_{cm}) - \theta_m) \right) \\ \geq 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} d(x_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}}. \end{array} \quad (2)$$

Et la fonction objective à minimiser est donnée par :

$$Ł = \sum_{n=1}^{N} \sum_{c \neq y_n} max\left(0\ ,\ 1 + \frac{1}{T_n} \sum_{t=1}^{T_n} \left(d(x_{nt}, \mu_{y_n m_{nt}}) + \theta_{m_{nt}} + log \sum_{m=1}^{M} exp(-d(x_{nt}, \mu_{cm}) - \theta_m) \right)\right). \quad (3)$$

3 Apprentissage des LM-dGMM restreint aux k-meilleures gaussiennes

3.1 Description du nouveau algorithme d'apprentissage

Malgré le fait que l'apprentissage des LM-dGMM soit plus rapide que celui des LM-GMM originaux de [4], sa complexité algorithmique reste encore trop élevée pour traiter de grands volumes de données. Afin de pouvoir utiliser ces modèles dans ce genre de scénario, nous proposons de réduire considérablement le nombre de contraintes à satisfaire dans (2), réduisant ainsi la complexité calculatoire de la fonction de perte et de son gradient (par rapport aux μ_{cm}).

Pour ce faire, nous utilisons le fait que les systèmes GMM de l'état-de-l'art effectuent la décision en ne tenant compte que des k-meilleures gaussiennes et non pas de l'ensemble des gaussiennes. Pour chaque vecteur x_{nt}, nous relaxons les contraintes à satisfaire en se limitant à présent aux k-meilleures gaussiennes de chaque classe c uniquement. Pour réduire d'avantage le temps de calcul et l'espace mémoire requis, nous exploitons une autre propriété : la correspondance qui existe entre les composantes des modèles GMM appris par adaptation MAP et celles de l'UBM [1]. Nous utilisons donc l'UBM pour sélectionner un ensemble S_{nt} unique des k-meilleures gaussiennes pour chaque vecteur x_{nt}, au lieu de $(C-1)$ ensembles. Nous avons donc une sélection $(C-1)$ fois plus rapide et moins demandeuse de mémoire (plus le nombre de locuteurs est grand plus le gain est important).

Les nouvelles contraintes à satisfaire sont définies par :

$$\begin{array}{l} \forall c \neq y_n, \\ \frac{1}{T_n}\sum_{t=1}^{T_n}\left(-log \sum_{m \in S_{nt}} exp(-d(x_{nt},\mu_{cm}) - \theta_m)\right) \\ \geq 1 + \frac{1}{T_n}\sum_{t=1}^{T_n} d(x_{nt},\mu_{y_n m_{nt}}) + \theta_{m_{nt}}. \end{array} \quad (4)$$

La fonction de perte à minimiser devient alors :

$$Ł = \sum_{n=1}^{N}\sum_{c \neq y_n} max\left(0\ ,\ 1 + \frac{1}{T_n}\sum_{t=1}^{T_n}\left(d(x_{nt},\mu_{y_n m_{nt}}) + \theta_{m_{nt}} + log \sum_{m \in S_{nt}} exp(-d(x_{nt},\mu_{cm}) - \theta_m)\right)\right). \quad (5)$$

Cette fonction est convexe est peut être minimisée par des algorithmes classiques d'optimisation non-linéaire tels que l'algorithme L-BFGS [10].

Durant la phase de test, nous utilisons le même principe pour accélérer le calcul des scores d'appariement (de vérification). Pour un segment de test donné $\{x_t\}_{t=1}^{T}$, et pour chaque vecteur x_t, nous utilisons l'UBM $\{\mu_{Um}, \Sigma_m, \theta_m\}$ pour sélectionner l'ensemble E_t des k-meilleures gaussiennes qui serviront à calculer le rapport moyen des log-vraisemblances :

$$LLR_{avg} = \frac{1}{T}\sum_{t=1}^{T}\left(log \sum_{m \in E_t} \exp(-d(x_t,\mu_{cm}) - \theta_m) - log \sum_{m \in E_t} \exp(-d(x_t,\mu_{Um}) - \theta_m)\right). \quad (6)$$

3.2 Traitement des données aberrantes

Nous adoptons la même stratégie que [4] pour traiter les données aberrantes. Les modèles GMM initiaux sont utilisés pour calculer les pertes accumulées dûe aux violations des contraintes dans l'eq. (4) :

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

Les segments aberrants sont ceux ayant de grandes valeurs de h_n. Nous pondérons donc les termes de perte par les poids $sw_n = min(1, \frac{1}{h_n})$, obtenant finalement la fonction de perte suivante :

$$Ł = \sum_{n=1}^{N} sw_n h_n. \quad (8)$$

Pour résumer, le nouvel algorithme d'apprentissage des modèles LM-dGMM consiste à :

- Initialiser chaque classe (locuteur) par un GMM appris par adaptation MAP,
- déterminer pour chaque vecteur de données, la composante du mélange ayant la plus grande probabilité a postériori,
- utiliser l'UBM pour sélectionner l'ensemble des k-meilleures gaussiennes associé à chaque vecteur de données,
- calculer les poids des segments,
- résoudre le problème d'optimisation non-linéaire définie par la fonction de perte de l'eq. (8)

$$min \quad Ł. \quad (9)$$

4 Résultats expérimentaux

Nos expérimentations sont effectuées sur la tâche de vérification du locuteur de la compagne d'évaluation NIST-SRE'2006 [6]. Les tests sont effectués sur les 349 locuteurs masculins de la condition principale (1conv4w-1conv4w). Les performances sont mesurées en terme de taux d'erreurs égales (EER) et de minimums de la fonction de coût de détection (minDCF) de NIST [11], et sont calculées sur une liste de 22123 unités d'évaluation (faisant appel à 1601 fichiers de test).

La paramétrisation est faite avec l'outil SPro [12]. Le signal de parole est filtré de manière à ne garder que la bande de fréquence [300-3400]Hz. Il est ensuite analysé localement à l'aide d'un fenêtrage temporel de type Hamming, des fenêtres glissantes de 20ms sont utilisées, à décalage régulier de 10ms. Des coefficients cepstraux LFCC (Linear Frequency Cepstral Coefficients) [13] sont calculés à partir d'un banc de 24 filtres à échelle linéaire. Ainsi le vecteur de paramètres se compose de 50 coefficients incluant 19 LFCCs, leurs dérivées premières, les 11 premières dérivées secondes et le delta-énergie. La phase de prétraitement comporte une normalisation CMVN [14] et une segmentation parole / non parole.

Les modèles GMM de base sont appris en utilisant l'outil état de l'art ALIZE/Spkdet [15]. L'adaptation MAP utilise un UBM des locuteurs masculins appris sur les données de NIST-SRE'2004. La technique (état de l'art) de compensation de la variabilité inter-sessions SFA [7, 8] est expérimentée dans les

tests, et utilise une matrice U de rang 40 estimée sur 2934 sessions de 124 locuteurs masculins de NIST-SRE'2004. Les GMM de base (avec ou sans SFA) sont utilisés comme initialisation dans l'algorithme d'apprentissage des LM-dGMM.

Le tableau 1 rassemble l'ensemble des résultats obtenus en terme de EER(%) et minDCF(x100), des modèles GMM et LM-dGMM à $M = 256$ et $M = 512$ gaussiennes, et pour un $k = 10$.

TAB. 1 – GMM (+SFA) vs LM-dGMM (+SFA)

Système	Configuration		EER	minDCF
GMM	M = 256	sans SFA	9.48	4.26
LM-dGMM	M = 256	sans SFA	8.97	3.97
GMM	M = 256	avec SFA	5.96	2.37
LM-dGMM	M = 256	avec SFA	5.58	2.29
GMM	M = 512	sans SFA	9.79	4.20
LM-dGMM	M = 512	sans SFA	9.66	4.13
GMM	M = 512	avec SFA	5.33	2.16
LM-dGMM	M = 512	avec SFA	5.02	2.18

Les résultats obtenus montrent clairement que notre algorithme d'apprentissage discriminant donne de meilleures performances que les modèles GMM génératifs, en réduisant les EERs et minDCFs dans les différentes configurations. Grâce à sa complexité algorithmique relativement faible, ces résultats suggèrent que les LM-dGMM pourraient constituer une bonne alternative à l'apprentissage génératif classique des GMM.

5 Conclusion

Nous avons présenté dans cette communication, un nouvel algorithme rapide d'apprentissage discriminant des GMM, restreint aux k-meilleures gaussiennes sélectionnées en utilisant le modèle du monde. Sa complexité réduite le rend utilisable dans des applications complexes, comme celles des compagnes d'évaluation NIST-SRE. Des tests effectués sur la tâche de vérification du locuteur de la compagne NIST-SRE'2006 montrent que nos modèles GMM à grande marge donnent de meilleurs résultats que les modèles génératifs classiques, ce qui rend notre approche intéressante et prometteuse. Ce travail ouvre de nouvelles perspectives, notamment la comparaison et la combinaison avec d'autres approches discriminantes, comme le système SVM à base de supervecteurs GMM.

Références

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

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

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

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

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

[6] The NIST Year 2006 Speaker Recognition Evaluation Plan, 2006, Online : http ://www.itl.nist.gov/iad/mig/tests/spk/2006/sre-06_evalplan-v9.pdf.

[7] Kenny, P. and Boulianne, G. and Ouellet, P. and Dumouchel, P., "Speaker and session variability in GMM-based speaker verification," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 4, pp. 1448–1460, 2007.

[8] Fauve, B.G.B. and Matrouf, D. and Scheffer, N. and Bonastre, J.-F. and Mason, J.S.D., "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.

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

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

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

[12] Gravier, G., "SPro : 'Speech Signal Processing Toolkit'," 2003, Online : https ://gforge.inria.fr/projects/spro.

[13] Davis, S. B., and Mermelstein, P., "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 28, Issue 4, pp. 357–366, 1980.

[14] Viikki, O., and Laurila, K., "Cepstral domain segmental feature vector normalization for noise robust speech recognition," Speech Communication, vol. 25, no. 1-3, pp. 133–147, 1998.

[15] Bonastre, et al., "ALIZE/SpkDet : a state-of-the-art open source software for speaker recognition," in Proc. of Odyssey-The Speaker and Language Recognition Workshop, 2008.