HMM évolutif pour les tâches de segmentation et d'indexation

Sylvain MEIGNIER*, Jean-François BONASTRE, Stéphane IGOUNET

LIA/CERI Université d'Avignon, Agroparc,
BP 1228, 84911 Avignon Cedex 9, France.
{sylvain.meignier, jean-francois.bonastre}@lia.univ-avignon.fr, stephane.igounet@univ-avignon.fr

Résumé – Cet article présente une méthode fondée sur un HMM pour la tâche d'indexation en aveugle de locuteurs. Cette méthode détecte et ajoute un à un les locuteurs dans un HMM évolutif (E-HMM). La solution proposée exploite l'ensemble des informations (locuteurs détectés) dès qu'elles sont disponibles. Le système proposé a été testé pour les tâches de « *N-segmentation* » lors de la campagne d'évaluation NIST 2001.

Abstract – This paper presents an iterative process for blind speaker indexing based on a HMM. This process detects and adds speakers one after the other to the evolutive HMM (E-HMM). The proposed solution exploits all the information (detected speakers) as soon as it is available. The proposed system was tested on *N-speaker* segmentation task of NIST 2001 evaluation campaign.

1 Introduction

La recherche des locuteurs intervenant au sein d'une conversation constitue une tâche essentielle pour l'indexation par le contenu de documents multimédia. Les systèmes d'indexation détectent les ruptures dans le signal sonore et regroupent les segments engendrés par ces ruptures en classes de sons homogènes.

Dans le domaine de la reconnaissance automatique du locuteur, l'indexation en aveugle de locuteur consiste à proposer pour un flux audio un descriptif définissant le nombre de locuteurs et leurs interventions.

Deux approches usuelles en indexation de locuteurs sont envisagées. La première approche, décrite notamment dans [1] et [2], repose sur deux phases séparées : la détection puis le regroupement par locuteurs. La seconde approche, proposée en particulier dans [3], est fondée sur l'utilisation de modèles de locuteur calculés par un système de reconnaissance automatique du locuteur (RAL). La détection des segments et l'attribution de ceux-ci aux différents locuteurs sont réalisées simultanément.

Aucune information *a priori* sur les locuteurs potentiels n'est utilisée dans ces deux approches. Cette caractéristique rend ces méthodes adaptées aux tâches d'indexation en aveugle.

Dans cet article, nous proposons une méthode apparentée à la deuxième approche. La conversation est modélisée par un modèle de Markov (proche de celui proposé dans [3]). Au cours du processus d'indexation, le modèle évolue à chaque nouvelle détection de classe de sons.

Le système proposé a été testé pour les tâches de « *N-segmentation* » lors de la campagne d'évaluation NIST 2001 [4].

* projet RAVOL : support financier du Conseil général de la région Provence Alpes Côte d'Azur et de DigiFrance.

2 Modèle de segmentation

2.1 Structure du modèle de segmentation

Le signal à indexer est constitué d'une séquence de vecteurs d'observation $O = (o_1, o_2, \ldots, o_T)$.

Les changements de classe sont représentés par un modèle de Markov caché ergodique (Hidden Markov Model : HMM). Chaque état modélise une classe de sons et les transitions représentent les changements entre les états. Dans cette application, chaque classe de sons correspond à un locuteur.

Le HMM λ est défini par (E, A, B) :

- $E = \{1, 2, \ldots, N\}$ un ensemble d'états. L'état i est associé au modèle de sons C_i de la classe $\mathcal{C}_i$.
- $A = \{a_{i,j}\}$ un ensemble de probabilités de transition entre les états.
- $B = \{b_i\}$. A chaque état i est associé un ensemble b_i de probabilités d'émission. On note $b_i(o_t)$ la probabilité du modèle C_i pour l'observation o_t.

Les probabilités de transition sont établies en fonction d'un ensemble de règles. Elles vérifient trois conditions :

$$\begin{cases} \forall i,\ a_{i,i} = \gamma \\ \forall (i,j),\ i \neq j,\ a_{i,j} = \frac{1-\gamma}{N-1} \\ 0 < \gamma < 1 \end{cases} \quad (1)$$

Nous ne disposons pas de connaissances *a priori* sur les changements de classe. Les probabilités $a_{i,j}$ correspondantes sont donc toutes égales.

2.2 Détection des classes et construction du modèle de segmentation

La construction du modèle de segmentation est réalisée par un processus itératif, qui détecte et ajoute à chaque itération i une classe de sons $\mathcal{C}_i$. Ce processus est réalisé en 4 étapes (FIG.1).

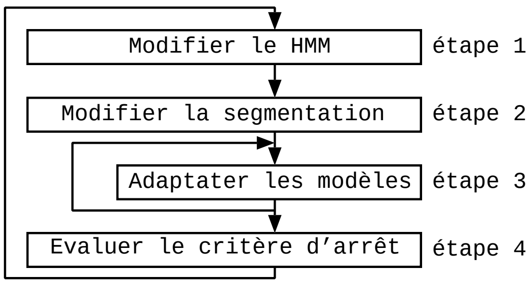

FIG. 1 – Diagramme du processus de segmentation

Note : lors de la description du processus, le numéro de l'itération sera porté en exposant.

À la première itération ($i = 1$) du processus (FIG.2, itération 1), le HMM $\lambda^1 = (E^1, A^1, B^1)$ est composé d'un seul état ($E^1 = \{1\}$), associé au modèle C_1^1 représentant l'ensemble du document. C_1^1 est appris sur la totalité des observations O.

Une première segmentation triviale $S^1 = (s_1^1, \ldots, s_T^1) = (\mathcal{C}_1, \ldots, \mathcal{C}_1)$ est générée. A chaque observation o_i correspond une étiquette $s_i^1 = \mathcal{C}_1$. Tous les observations appartiennent à la classe $\mathcal{C}_1$. S^1 sera remise en cause à l'itération suivante.

Dans les itérations suivantes du processus (FIG.2, itération 2 & 3), les différentes étapes pour $i > 1$ sont :

étape 1 : Un nouvel état i est ajouté dans l'ensemble E^{i-1} ($E^i = E^{i-1} \cup \{i\}$). Les probabilités de transition sont calculées suivant les règles définies dans l'équation 1. On obtient le nouveau HMM $\lambda^i = (E^i, A^i, B^{i-1})$.

étape 2 : Le modèle C_i^i est construit à partir d'une séquence de longueur fixe d'observations $(o_r, o_{r+1}, \ldots, o_{r+t})$ étiquetée $\mathcal{C}_1$. Cette séquence est sélectionnée telle que :

$$\left\{ \begin{array}{l} r = \underset{j \in L}{\mathrm{ArgMax}} \left(\prod_{k=j}^{j+t} b_1^i(o_k) \right) \\ L = \{j | s_j^{i-1} = s_{j+1}^{i-1} = \ldots = s_{j+t}^{i-1} = \mathcal{C}_1\} \end{array} \right. \quad (2)$$

ou $r \in L$ est le rang de la première observation de la séquence $(o_r, o_{r+1}, \ldots, o_{r+t})$, qui maximise le produit des probabilités $\{b_1^i(o_r), b_1^i(o_r + 1), \ldots, b_1^i(o_r + t)\}$ calculé à partir du modèle C_1^i.

L'indexation S^i est générée : la séquence $(o_r, o_{r+1}, \ldots, o_{r+t})$ est affectée à la classe $\mathcal{C}_i$.

$$\left\{ \begin{array}{l} s_j^i = s_j^{i-1} \; \forall j \notin \{r, \ldots, r+t\} \\ s_r^i = s_{r+1}^i = \ldots = s_{r+t}^i = i \end{array} \right. \quad (3)$$

étape 3 : Dans cette phase, elle même itérative, le processus adapte les paramètres du HMM λ^i :

a - Pour chaque $k \in \{1, \ldots, i\}$, le modèle C_k^i est adapté en fonction des observations qui lui ont été affectées dans l'indexation S^i.

b - L'ensemble des probabilités d'émission B^i est calculé.

c - L'algorithme de Viterbi permet de calculer une nouvelle version de S^i pour obtenir l'alignement optimal par rapport au HMM actuel. On a la probabilité du chemin :

$$P(S^i | A^i, B^i, O) = b^i_{s_1^i}(o_1) \times \prod_{j=2}^{T} (a^i_{s_{j-1}^i, s_j^i} \times b^i_{s_j^i}(o_j)) \quad (4)$$

Si un gain est observé, entre deux itérations de l'étape 3, le processus reprend en 3-a.

étape 4 : Le critère d'arrêt est évalué : si

$$P(S^i | A^i, B^i, O) > P(S^{i-1} | A^i, B^{i-1}, O) \quad (5)$$

alors une nouvelle itération commence à l'étape 1.

Note : la probabilité de S^{i-1} est réestimée avec les probabilités de transition du modèle λ^i, pour supprimer au niveau des transitions l'influence du nombre d'état.

3 Système de RAL

Les modèles de sons employés et l'ensemble des probabilités d'émission sont calculés par le système de reconnaissance du locuteur AMIRAL, développé au LIA [5][6][7]. La paramétrisation acoustique est calculée grâce au module SPRO développé par le consortium ELISA [6]. Cette paramétrisation est composée de 16 coefficients cepstraux et de leurs dérivées, calculés toutes les 10ms sur une fenêtre de 20ms.

Les classes de son sont modélisées par des mixtures de gaussiennes (Gaussian Mixture Model : GMM) à 128 composantes à matrice de covariance diagonale [8]. Les modèles GMM sont adaptés à partir d'un modèle du monde appris sur un corpus séparé (en utilisant l'algorithme EM-ML [9]). Le modèle de sons C est adapté dans un premier temps sur une séquence d'observations d'une durée fixe de 3 secondes (§ 2.2 - étape 2).

Puis, le modèle C est adapté à partir des segments étiquetés par la classe $\mathcal{C}$ (§ 2.2 - étape 3). La procédure d'adaptation est fondée sur la méthode du *maximum a posteriori* (MAP [6]).

Dans le cadre de cette application, seules les moyennes du modèle sont adaptées ; le modèle utilise les poids et la matrice de covariance du modèle du monde. Pour chaque gaussienne g, la moyenne μ_g du modèle C est une combinaison linéaire de la moyenne estimée $\widehat{\mu}_g$ et de la moyenne correspondante μ_g^W du modèle du monde W.

$$\mu_g = \alpha \mu_g^W + (1 - \alpha) \widehat{\mu}_g, \alpha > 0 \quad (6)$$

Les probabilités d'émission (rapports de vraisemblances de C et W) sont calculées sur des blocs de longueur fixe de 0.3 seconde.

4 Expériences

4.1 Ensembles de données

Le corpus utilisé provient de la tâche « *N-segmentation* » de NIST[4]. Il est composé de 500 conversations télépho-

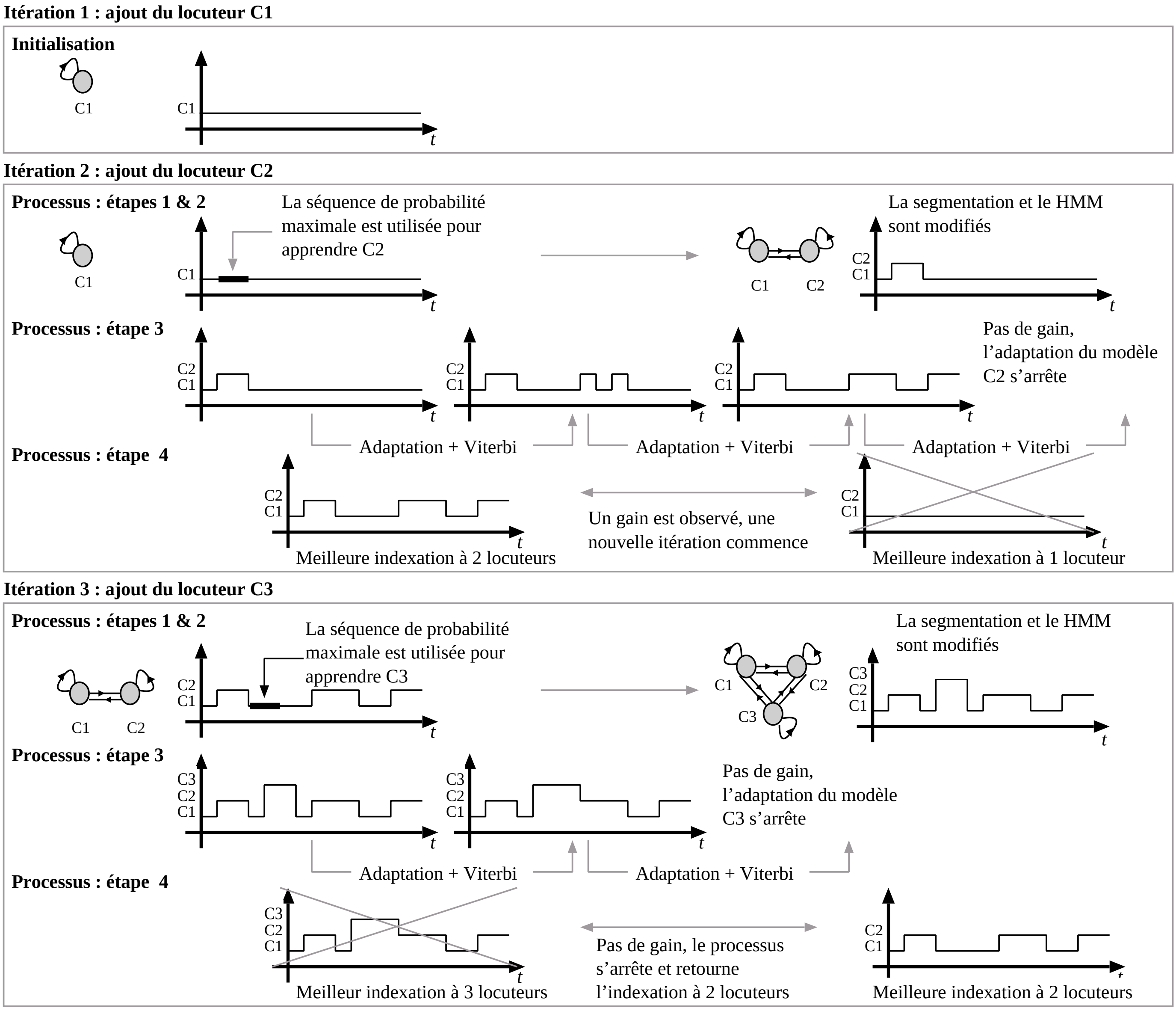

FIG. 2 – Exemple de segmentation pour un test contenant 2 locuteurs

niques provenant du corpus CALLHOME (multi-locuteur, multi-langue). Les tests du corpus sont de durées variables ($<$ 10 minutes) et ils couvrent 6 langues. Le nombre exact de locuteurs est inconnu (mais $<$ 10). Lors des expériences, ce corpus est divisé en deux sous-ensemble de 250 tests nommés respectivement *Dev* et *Eva*.

NIST fournit d'autre part un corpus de développement *train_ ch*, composé de 48 conversations extraites de CALLHOME. Cet ensemble sert de données d'apprentissage pour le modèle du monde *wld_ ch*.

Un second ensemble de développement *train_ sb* est utilisé pour l'apprentissage du modèle d'un monde *wld_ sb*. Ce corpus est composé de 472 tests provenant de Switchboard 2 ($\simeq$ 100 locuteurs), et présente l'interêt d'être de taille plus importante.

4.2 Expériences et paramètres

Les expériences, réalisées après les évaluation NIST, estiment l'influence du paramètre α (Eq. 6) lors de l'apprentissage MAP en fonction du modèle du monde (*wld_ ch* ou *wld_ sb*). Pour référence, le résultat d'une segmentation triviale composée d'un seul segment par test est donné.

Les résultats sont obtenus avec les paramètres suivants :
- Les probabilités de transition sont estimées avec $\gamma = 0.6$ (Eq. 1).
- Les paramètres MAP sont :

$$\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$$

4.3 Résultats de développement et NIST 2001

Résultats de développement : Le score de segmentation NIST [4] est calculé sur les segments de référence. Seul les segments où un locuteur unique parle sont pris en compte dans le score. Ce score correspond à une erreur de segmentation.

Les résultats de développement de la figure 3 montrent le score de segmentation obtenu avec les différents modèles du monde sur le corpus *Dev*.

Quand le poids d'adaptation α est proche de 1, le score devient équivalent au résultat de la segmentation triviale (score de 38%). Le système attribute majoritairement les observations à une classe unique.

Le modèle *wld_ ch* dégrade le score de segmentation par rapport à *wld_ sb*, bien que les données d'apprentissage *train_ sb* soient très différentes des données *train_ ch*. L'ensemble *train_ ch*, étant un ensemble de taille réduite par rapport à *train_ sb*, il ne généralise pas suffisamment les données du corpus CALLHOME.

Les meilleurs résultats observés sur *Dev* sont proches des résultats obtenus sur *Eva* (TAB.1).

TAB. 1 – Meilleurs résultats sur Dev et Eva

monde	α	score (*Dev*)	score (*Eva*)
wld_ sb	0	24.01%	23.42%
wld_ ch	0.3	25.50%	25.17%

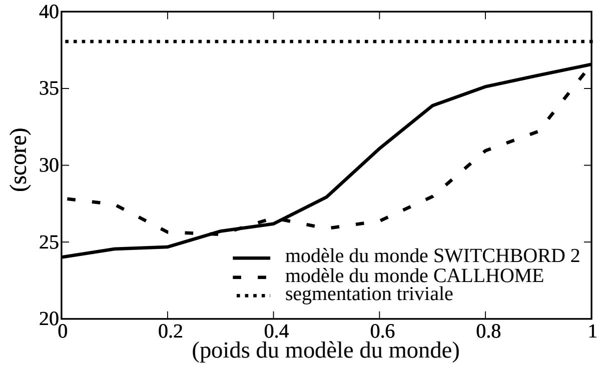

FIG. 3 – Score de segmentation NIST (%) : influence du paramètre α

Résultat NIST 2001 : Le système présenté à NIST utilise le modèle du monde *wld_ sb*. Il a obtenu sur le corpus d'évaluation NIST (correspondant à *Dev+Eva*) un score de 24% (FIG.4). Le système proposé est adapté pour la segmentation de fichier multi-locuteur quelque soit le nombre de classes à détecter. Le score du système est stable bien que le modèle du monde soit uniquement appris sur des données de langue anglaise alors que les tests comportent 6 langues.

5 Conclusion

Dans cet article, le système de segmentation utilise un modèle de Markov évolutif pour modéliser la conversation et pour déterminer automatiquement les classes présentes dans les signaux sonores. L'approche est basée sur un algorithme itératif qui détecte et ajoute les modèles de classe un à un. A chaque étape, une indexation est proposée, en fonction de l'ensemble des connaissances disponibles. Cette indexation est remise en cause à l'itération suivante jusqu'à l'optimal.

Au vu des résultats, le système se comporte de manière satisfaisante malgré la difficulté de la tâche (parole spontanée téléphonique). Les expériences montrent que l'apprentissage MAP est adapté pour des durées courtes d'apprentissage. Les travaux futurs porteront sur la modélisation des durées d'intervention et sur le choix du critère d'arrêt.

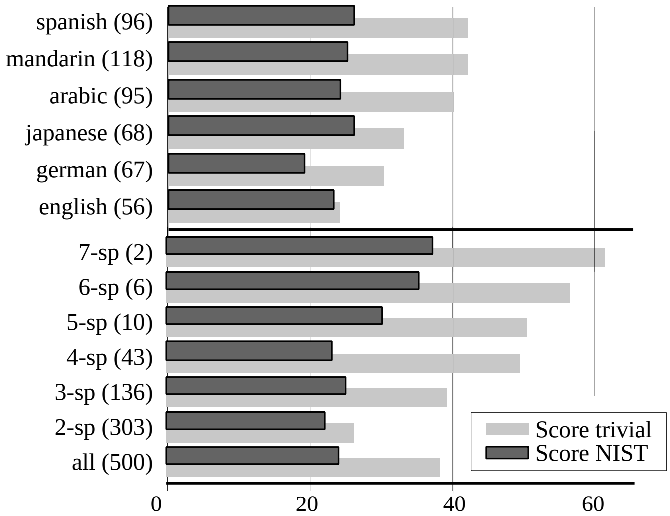

FIG. 4 – NIST 2001 *Dev+Eva* : score par rapport à la langue du test, score par rapport au nombre de locuteurs. Le nombre de tests est donné pour chaque condition

Références

[1] P. Delacourt, D. Kryze, C.J. Wellekens. *Use of second order statistic for speaker-based segmentation*, EUROSPEECH, 1999.

[2] H. Gish, H-H Siu, R. Rohlicek. *Segregation of speakers for speech recognition and speaker identification*, ICASSP, pages 873-876, 1991.

[3] K. Sönmez, L. Heck, M. Weintraub, *Speaker tracking and detection with multiple speakers*, EUROSPEECH, 1999.

[4] The NIST Year 2001 Speaker Recognition Evaluation Plan, http ://www.nist.gov/speech/tests/spk/2001 /doc/2001-spkrec-evalplan-v53.pdf.

[5] C. Fredouille, J.-F. Bonastre, T. Merlin, *AMIRAL : a block-segmental multi-recognizer approach for Automatic Speaker Recognition*, Digital Signal Processing, Vol.10, Num.1-3, pp.172-197 Janvier-Avril 2000.

[6] Elisa Consortium, *Overview of the ELISA consortium research activities*, Odyssey, 2001.

[7] Besacier L., Bonastre J.F., *Subband approach for automatic-speaker recognition*, European Journal Signal Processing, Elsevier, in Special Issue on Emerging Techniques for Communication Terminals, Vol 80, pp 1245-1259, 2000.

[8] D. A. Reynolds, *Speaker identification and verification using gaussian mixture speaker models*, Speech Communication, pp 91-108, Aug. 1995.

[9] D. Dempster, N. Larid, D. Rubin, *Maximum likelihood from incomplete data via EM algorithm*, J. Roy. Stat. Soc., Vol. 39, pp 1-38, 1977.