Nouveau noyau de séquences pour la vérification du locuteur

Jérôme Louradour, Khalid Daoudi

Institut de Recherche en Informatique de Toulouse - CNRS UMR 5505
118, route de Narbonne, 31062 Toulouse cedex 04, France
louradou@irit.fr, daoudi@irit.fr

Résumé – En utilisant la théorie des Espaces de Hilbert à Noyau Reproduisant, nous concevons un nouveau noyau de séquences, qui mesure la similarité entre deux séquences d'observations. Nous appliquons ce noyau à une tâche de vérification du locuteur (campagne d'évaluation NIST 2004). Les résultats montrent qu'incorporer notre nouveau noyau de séquences dans une architecture SVM non seulement fournit des résultats bien meilleurs qu'un classifieur UBM-GMM de base, mais aussi donne de meilleures performances que le classifieur utilisant un noyau GLDS (Generalized Linear Discriminant Sequence kernel). De plus, notre noyau opère dans un espace de plus faible dimension, tout en permettant un large choix de noyaux.

Abstract – Using the framework of Reproducing Kernel Hilbert Spaces, we develop a new sequence kernel that measures similarity between sequences of observations. We then apply it to a text-independent speaker verification task using the NIST 2004 Speaker Recognition Evaluation database. The results show that incorporating our new sequence kernel in an SVM training architecture not only yields performance significantly superior to those of a baseline UBM-GMM classifier but also outperforms the Generalized Linear Discriminant Sequence (GLDS) Kernel classifier. Moreover, our kernel maps to a relatively low dimensional feature space while allowing a large choice for the kernel function.

1 Introduction

La vérification du locuteur est un problème de classification binaire de séquences. Pour ce genre de problème, les méthodes à noyaux, comme les Machines à Vecteurs Supports (SVM), donnent des performances égales ou meilleures que les autres classifieurs. Alors que les algorithmes de base de ces méthodes sont maintenant bien maîtrisés [1], et que leurs propriétés ont été largement étudiées, trouver la façon optimale de représenter les données en entrée de ces algorithmes reste un problème ouvert.

Dans le cas de la vérification du locuteur, la décision est binaire, et les entités à classifier sont des séquences (dans nos expériences, séquences de vecteurs acoustiques). En pratique, les données d'apprentissage du locuteurs (classe +1), en plus d'être bruitées, sont éparpillées au sein même du nuage des données du monde, que l'on assimilera aux données d'apprentissage imposteurs (classe -1). La géométrie du problème est de surcroît complexifiée par le biais éventuel entre les conditions d'apprentissage et celles de test (combinés téléphoniques, canaux de transmissions, fond sonore).

Pour appliquer les SVM à la tâche de vérification, une idée simple consisterait à apprendre pour chaque locuteur une frontière de décision (non linéaire) dans l'espace des vecteurs d'observation, et moyenner les sorties sur chaque vecteur en phase de test, comme il a été fait dans [2] pour une tâche d'identification. Mais à cause des raisons invoquées dans le paragraphe précédent, cette méthode donne de mauvaises performances en vérification du locuteur. Aussi, la finalité du problème étant de classifier des séquences, il est plus adéquat d'optimiser les critères d'apprentissage au niveau des séquences plutôt que des vecteurs.

Plusieurs études récentes ont porté sur la conception de noyaux pour les séquences, de tels noyaux visant à quantifier la similarité entre séquences vis-à-vis d'un problème donné. On peut distinguer trois types d'approches :

- Une première tendance pour comparer deux séquences consiste à entraîner deux modèles statistiques à partir de chacune d'entre elles, et à définir le noyau comme la similarité entre les deux distributions estimées (divergence de Kullback-Leibler [3], affinité de Bhattacharyya [4], distance du χ_2, etc.). Dans notre cas, la courte durée des séquences d'apprentissage ne permet pas d'estimer un modèle de façon suffisamment fiable.
- Une seconde tendance consiste à travailler dans l'espace des scores probabilistes, ou encore l'espace dérivé via le noyau de Fisher [5, 6]. Ces méthodes manquent d'efficacité : elles restent lourdes à mettre en œuvre, et n'améliorent que modérément les performances par rapport aux systèmes fondés sur une décision bayésienne.
- Une autre stratégie [7] revient à apprendre un modèle sur une séquence et à attribuer un score en conséquence sur une autre pour évaluer le noyau entre ces deux séquences. Dans le cas d'une modélisation discriminante par expansion polynomiale, apprendre sur une séquence A et tester sur une séquence B revient au même que d'apprendre sur B et tester sur A. Ceci conduit à l'élaboration d'un noyau symétrique. Cette catégorie de méthodes fournit des résultats très prometteurs : les performances sont meilleures que les systèmes à base de modèles génératifs, pour une complexité moindre.

Dans cette dernière approche, le noyau de séquence se calcule en projetant explicitement les séquences dans un espace de dimension fixe D, via une expansion polynomiale, et en appliquer un produit scalaire après normalisation. En notant d la dimension des vecteurs d'entrée, et k l'ordre de l'expansion polynomiale, la dimension du *feature space* est $D = \frac{(k+d)!}{K!d!}$, Pour des valeurs de k supérieures à 3, elle devient trop grande (*e.g.* $D = 20.475$ pour $d = 24$ et $k = 4$) et rend le problème intraitable. Le choix du noyau est de ce fait très limité. C'est pour combler ce manque de flexibilité que nous avons conçu un nouveau noyau de séquence, en adoptant la même philosophie.

2 Conception du noyau

Cette section introduit le raisonnement théorique qui conduit à la formulation du nouveau noyau de séquence.

2.1 Apprentissage via l'espace de Hilbert à Noyau Reproduisant

Considérons un corpus d'apprentissage $(\tau_n, s_n)_{n=1...N}$, constitué de vecteurs $\tau_n \in \Re^d$ et d'étiquettes binaires $s_n \in \{0, 1\}$ qui valent 0 pour un ensemble fixé de données du monde $C = (c_m)_{m=1...M}$ éparses, et 1 pour les vecteurs d'une séquence $A = (a_t)_{t=1...T_A}$ produite par un locuteur cible donné. On cherche une fonctionnelle $f : \Re^d \to \Re$ qui minimise la fonction objectif :

$$\min_{f \in \mathcal{H}} \sum_{n=1}^{N} \parallel f(\tau_n) - s_n \parallel_2 \tag{1}$$

où $\mathcal{H}$ est l'espace de recherche de la solution $\hat{f}$.

Une sous-classe importante de problèmes de la forme (1) sont générés par des fonctions noyau définies positives $K(x, y)$. Le sous-espace de fonctions correspondant, $\mathcal{H}_K$, est appelé *espace de Hilbert à noyau reproduisant*. Supposons que K ait une expansion de la forme :

$$K(x, y) = \sum_{i=1}^{D} \gamma_i^2 \Psi_i(x) \Psi_i(y) = \langle \gamma\Psi(x), \gamma\Psi(y) \rangle \tag{2}$$

avec $\gamma = diag(\gamma_1, ..., \gamma_D)$ et $\Psi : \Re^d \to \Re^D$ une fonction à haute dimension (D peut être infini). Alors tout élément de $\mathcal{H}_K$ est de la forme $f(x) = \sum_{i=1}^{D} \epsilon_i \Psi_i(x)$.

Dans [8], G. Whaba montre que la solution à (1) est de la forme :

$$f(x) = \sum_{n=1}^{N} \omega_n K(x, \tau_n) \tag{3}$$

Dans la suite, notre *mapping* de séquence pour le calcul du noyau (6,8,9) sera basé sur les fonctions de base $x \mapsto K(x, \tau_n)$. En vue de concevoir un système stable et efficace, il est préférable de concevoir un noyau qui soit indépendant du locuteur cible. C'est pourquoi nous cherchons en fait les solutions sous la forme :

$$f(x) = \sum_{n=1}^{M} \omega_m K(x, c_m) \tag{4}$$

La résolution des équations normales avec quelques approximations (*cf.* [9] pour plus de détails) donne la solution :

$$\hat{\omega}_A = [\hat{\omega}_1, ..., \hat{\omega}_M]^{\mathrm{T}} = M.{\mathbf{K}_C}^{-2} \overline{\varphi_{\mathbf{C}}}(A) \tag{5}$$

où $\mathbf{K}_C = (K(c_m, c_n))$ est la matrice de similarité symétrique, et où l'on définit les transformations :

$$\begin{cases} \overline{\varphi_{\mathbf{C}}}(x_1, ..., x_T) = \frac{1}{T} \sum_{t=1}^{T} \varphi_{\mathbf{C}}(x_t) \\ \varphi_{\mathbf{C}}(x) = [K(x, c_1), ..., K(x, c_M)]^{\mathrm{T}} \end{cases} \tag{6}$$

2.2 Similarité entre deux séquences

Supposons avoir appris à partir d'une séquence A un modèle discriminant de la forme (5). La fonction de similarité entre un vecteur x et la séquence A est donnée par :

$$\hat{f}(x) = \langle \hat{\omega}_A, \varphi_C(x) \rangle \tag{7}$$

Pour étendre cette mesure à une séquence $B = (b_t)_{t=1...T_B}$, nous moyennons la similarité de chacun de ses vecteurs :

$$\begin{aligned} similarité(B|A) &= \frac{1}{T_B} \sum_{t=1}^{T_B} \hat{f}(b_t) \\ &= M.\overline{\varphi_{\mathbf{C}}}(B)^{\mathrm{T}} {\mathbf{K}_C}^{-2} \overline{\varphi_{\mathbf{C}}}(A) \end{aligned}$$

Au facteur multiplicatif M près, cela conduit à notre nouveau noyau de séquence symétrique :

$$\kappa(A, B) = \langle \overline{\Phi_C}(A), \overline{\Phi_C}(B) \rangle \tag{8}$$

où l'on définit le *mapping* de séquence :

$$\overline{\Phi_C}(x_1, ...x_T) = {\mathbf{K}_C}^{-1} \overline{\varphi_{\mathbf{C}}}(x_1, ..., x_T) \tag{9}$$

Dans [9], nous montrons que ce *mapping* est équivalent à une projection, dans l'espace de dimension D défini par Ψ dans (2), de l'expansion moyenne $\frac{1}{T} \sum_{t=1}^{T} \Psi(x_t)$, sur la base des expansions des vecteurs du monde $(\Psi(c_m))_{m=1...M}$.

3 Implémentation du noyau

L'implémentation du noyau de séquences (8) se fait en projetant chaque séquence dans un espace de dimension M, et en calculant un produit scalaire. Ceci est délicat à mettre en œuvre lorsque M est trop grand. C'est pourquoi nous avons supposé en 2.1 que les données imposteurs c_m étaient éparses. Cependant, en pratique, la quantité de données du monde disponibles est très grande. Dans ce qui suit, l'ensemble C est obtenu par quantification vectorielle de données du monde, et les c_m sont des vecteurs *codebook* représentatifs de ces données.

Le but de cette section est de présenter comment appliquer notre noyau de séquence, pour un problème de vérification du locuteur.

3.1 Notion de séquence

Dans notre problématique, une séquence est définie comme un ensemble de vecteurs produits par un même locuteur dans les mêmes conditions d'enregistrement.

Des expériences ont montré que même si une seule séquence était disponible par locuteur, il était inutile de la diviser en plusieurs. Un tel artifice n'améliore pas les performances. Par contre, si l'on dispose de plusieurs séquences provenant de différentes sessions d'enregistrement, alors il est préférable de les conserver distinctes, afin de garder l'information sur la variabilité due aux conditions d'enregistrement.

3.2 Mapping de séquence

Le *mapping* de séquence $\overline{\Phi_C}$ est entièrement déterminé par un ensemble de vecteurs *codebook* C et une fonction K satisfaisant les conditions de Mercer. L'implémentation du *mapping* d'une séquence $X = (x_1, ..., x_T)$ est montrée dans la figure 1.

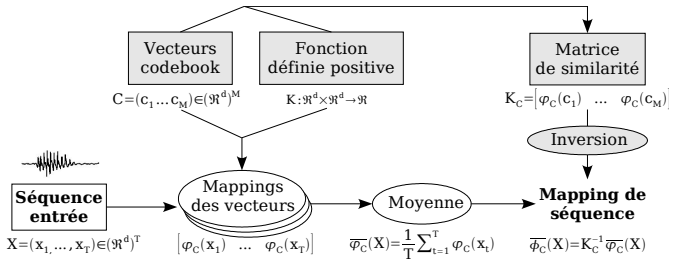

FIG. 1: *Implémentation du mapping de séquences. Les blocs grisés désignent les étapes préliminaires (phase de développement).*

3.3 Apprentissage de modèle discriminant

Une fois le *mapping* défini, nous pouvons pré-calculer les *mappings* des séquences imposteurs d'un corpus du monde, et entraîner un modèle SVM par locuteur cible, avec la procédure *un-contre-tous* schématisée dans la figure 2. Dans nos expériences, Les modèles SVM ont été estimés avec SVM Torch [10]. Il est important de noter que les composantes du *mapping* sont normalisées de façon à avoir une variabilité unitaire sur chaque entrée. La normalisation est donnée par :

$$\overline{\Phi_C}(X) \mapsto \frac{\overline{\Phi_C}(X) - \mu}{\sigma} \tag{10}$$

où μ et σ sont les estimations respectives de la moyenne et l'écart-type, sur le corpus du monde, des vecteurs projetés $\overline{\Phi_C}(X_{imp})$. Cette normalisation est une précaution classique pour les SVMs, qui ne sont pas invariants aux transformations linéaires.

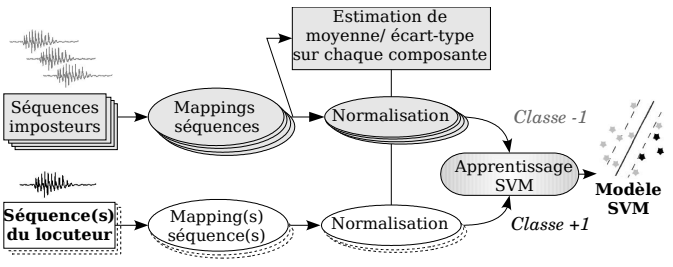

FIG. 2: *Apprentissage de modèle locuteur par SVM*

3.4 Test d'une séquence

Modulo un seuil de décision, la sortie d'une SVM sur une séquence Y est de la forme :

$$score(Y) = \sum_i \alpha_i y_i \langle \overline{\Phi_C}(Y), \overline{\Phi_C}(S_i) \rangle \tag{11}$$

où les (α_i) sont des poids positifs appris, les (S_i) sont les séquences d'apprentissage, et les $y_i = \pm 1$ sont les étiquettes correspondantes ($+1$ pour les séquences du locuteur cible, -1 pour les séquences du monde).

Étant donné la linéarité du produit scalaire, le calcul du score peut être simplifié :

$$score(Y) = \langle \overline{\Phi_C}(Y), \sum_i \alpha_i y_i \overline{\Phi_C}(S_i) \rangle = \langle \overline{\Phi_C}(Y), \Omega_{loc} \rangle \tag{12}$$

Comme dans [7], compacter les séquences support en un seul modèle Ω_{loc} permet d'économiser de l'espace mémoire pour le stockage des modèles locuteur, et de réduire les temps de calcul en phase de test.

Finalement, la décision binaire *locuteur/imposteur* pour une séquence Y est prise en comparant $score(Y)$ à un seuil pré-déterminé.

4 Expériences

4.1 Données de développement et de test

Nous avons appliqué notre nouveau noyau de séquences à la campagne d'évaluation de reconnaissance du locuteur NIST 2004, dans les conditions standards [11]. Dans ces conditions, une séquence contenant plus ou moins deux minutes de parole est disponible pour entraîner chaque modèle locuteur. Pour le corpus du monde, nous avons utilisé un millier de séquences extraites des données de la campagne NIST 2001.

4.2 Pré-traitement

A partir d'une séquence de parole, 12 MFCC sont extraits sur des fenêtres de Hamming de 16 ms, avec un pas régulier de 10 ms. Les 12 coefficients dérivés et la dérivée de l'énergie en échelle logarithmique sont aussi rajoutés. Un détecteur d'activité vocale rejette alors les zones de faible énergie.

Enfin, les vecteurs de dimension 25 ainsi calculés sont normalisés par *feature warping* [12], sur une fenêtre glissante de 300 échantillons. Une telle normalisation a pour but de réduire l'effet combiné du bruit additif et des distortions dues aux canaux de transmission.

4.3 Systèmes de référence

Pour valider le nouveau système que nous avons conçu, nous le comparons à deux systèmes de références à partir de mêmes données de développement et de test.

Le premier, qui a été présenté par l'IRIT à l'évaluation NIST 2004, est un système UBM-GMM [13]. Il estime deux modèles Mélanges de Loi Gaussiennes, pour chacun

des genres homme/femme, en appliquant un algorithme EM sur les données du monde. Chaque modèle locuteur est alors dérivé du modèle du monde correspondant en adaptant les vecteurs moyennes avec un critère MAP. En phase de test, le score d'une séquence est la moyenne des rapports de vraisemblance.

Le second est fondé sur le noyau GLDS décris dans [7]. Le principe est similaire à notre approche, mais le *mapping* de séquence est différent. Il s'agit d'une expansion polynomiale moyenne, suivi d'une normalisation dont les paramètres sont aussi estimés à partir des données du monde. L'ordre polynomial maximal est fixé à 3 et la taille du *mapping* est alors $\frac{(25+3)!}{25!3!} = 3276$.

4.4 Résultats

Les courbes DET relatives aux deux systèmes de référence et à un système basé sur notre nouveau noyau de séquence, sont tracés dans la figure 3 (chaque point de la courbe correspond à un seuil de décision). La fonction de coût de détection considérée, à minimiser, est celle définie par NIST SRE [11], comme somme pondérée des probabilités de faut rejet et de fausse acceptation (*resp.* P_{fr} and P_{fa}) :

$$DCF = (0.1 \times P_{fr}) + (0.9 \times P_{fa}) \qquad (13)$$

Pour notre système, nous avons choisi un noyau polynomial de degré 7 : $K(x, y) = (1+\langle x, y\rangle)^7$. Un noyau gaussien donne des performances similaires. $M = 2048$ vecteurs *codebook* ont été estimés par quantification vectorielle sur les données du monde. Nos expériences ont montré qu'augmenter le nombre M améliorait les performances en général.

On peut voir que notre système donne de meilleures performances que les autres sur toutes les régions de la courbe DET. Des expériences sur l'évaluation NIST 2003, avec exactement les mêmes choix techniques, ont confirmé cette tendance. Par rapport à méthode utilisant un noyau GLDS, notre *mapping* est de dimension inférieure.

5 Conclusion

Nous avons introduit un nouveau noyau de séquences à l'origine d'un nouveau système SVM pour la vérification du locuteur. Toutes les expériences que nous avons menées ont montré que ce système fournit des résultats bien meilleurs qu'un classifieur UBM-GMM. De plus, il donne des performances comparables au classifieur SVM avec noyau GLDS, tout en travaillant dans un espace de dimension plus faible. Aussi, la flexibilité de notre nouveau noyau offre de belles perspectives pour des améliorations futures, et peut être appliqués à d'autres tâches de classification.

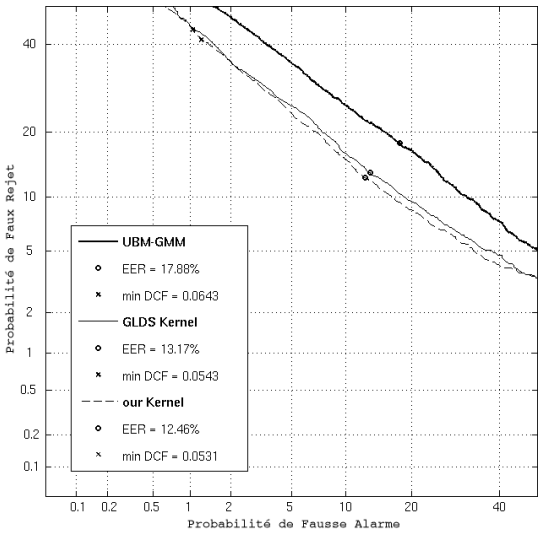

FIG. 3: *courbes DET - Comparaison de la nouvelle approche avec deux systèmes de références. Les cercles indiquent les EER, les croix montrent les points de fonctionnement correspondant au minimum de la fonction coût définie en (13)*

Références

[1] B. Schölkopf and A.J. Smola, *Learning with kernels : Support Vector Machines, regularization, optimization and beyond*, MIT Press, 2001.

[2] M. Schmidt and H. Gish, "Speaker identification via support vector machines," in *Proc. ICASSP*, 1996.

[3] P. Moreno and P. Ho, "A new svm approach to speaker identification and verification using probabilistic distance kernels," in *Proc. Eurospeech*, 2003.

[4] R. Kondor and Jebara T., "A kernel between sets of vectors," in *Proc. ICML*, 2003.

[5] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," *Advances in Neural Information Processing Systems 11*, 1998.

[6] V. Wan and S. Renals, "Speaker verification using sequence discriminant support vector machines," *IEEE Trans. on Speech and Audio Processing*, 2004.

[7] W.M. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition," in *Proc. ICASSP*, 2002.

[8] G. Wahba, *Applied Mathematics*, vol. 59, chapter Spline Models for Observational Data, CBMS-NSF Regional Conference Series, 1990.

[9] J. Louradour, "a new sequence kernel and its application to speaker verification," Irit research report, www.irit.fr/∼jerome.louradour/papers/, 2005.

[10] R. Collobert and Bengio, "Svmtorch : Support vector machines for large-scale regression problems," *Journal of Learning Machine Research*, 2001.

[11] "Nist speaker recognition 2004 evaluation plan," http://www.nist.gov/speech/tests/spk/2004/.

[12] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Speaker Odyssey*, 2001.

[13] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, 2000.