Généralisation du noyau GLDS pour la Vérification du locuteur par SVM

Jérôme Louradour [(1)]*, Khalid Daoudi* [(1)] *et Francis Bach* [(2)]

[(1)] IRIT, CNRS UMR 5505, Université Paul Sabatier, Toulouse, France
[(2)] Centre de Morphologie Mathématique, Ecole des Mines de Paris, Fontainebleau, France
Mél : louradou,@irit.fr, daoudi@irit.fr, francis.bach@mines.org

ABSTRACT

The Generalized Linear discriminant Sequence (GLDS) kernel provides good performance in SVM speaker verification in NIST SRE (Speaker Recognition Evaluation) evaluations. It is based on an explicit mapping of each sequence to a single vector in a feature space using polynomial expansions. Because of practical limitations, these expansions has to be of degree less or equal to 3. In this paper, we generalize the GLDS kernel to allow not only any polynomial degree but also any expansion (possibly infinite dimensional) that defines a Mercer kernel (such as the RBF kernel). We conceive a new kernel, and makes it tractable using a method of data reduction adapted to kernel methods : the Incomplete Cholesky Decomposition (ICD). We present experiments on NIST SRE database, that show good perspective for our new approach.

1. INTRODUCTION

Les algorithmes de classification Machines à Vecteurs de Support (SVM) sont aujourd'hui considérés comme une des méthodes les plus performantes pour de nombreux problèmes réels de classification et de régression. A l'origine, ils ont été conçus pour construire une fonction discriminante permettant de séparer au mieux des régions complexes, dans des problèmes de classification binaire. En cela ils constituent une alternative intéressante aux approches génératives classiques MMG (Mélanges de Modèles Gaussiens) pour la vérification du locuteur. Alors que les modèles génératifs sont particulièrement adaptés lorsque les données à classifier sont des séquences de vecteurs, de longueurs variables, un des principaux défis pour appliquer une approche discriminante à la biométrie à partir d'un extrait de parole est de l'adapter au traitement de données séquentielles.

Une des manières intuitives d'appliquer un SVM à la vérification du locuteur serait d'utiliser la même démarche qu'avec les modèles génératifs, c'est-à-dire d'apprendre des modèles discriminants dans l'espace des vecteurs acoustiques et de combiner les scores en phase de test pour décider de la classe d'une séquence. Mais malgré les récents efforts pour faire correspondre les scores SVM à des probabilités a posteriori [10], permettant ainsi de les combiner de manière naturelle via le théorème de Bayes, un problème en reconnaissance du locuteur reste de pouvoir exploiter des corpus d'apprentissage importants afin de caractériser au mieux les classes (locuteur / reste du monde) fortement multimodales. En effet, la complexité des algorithmes d'apprentissage empêche les SVM de profiter d'une affluence de données de développement, à moins d'avoir recours à des méthodes de réduction de données. Les approches par clustering [7] ne donnent pour l'instant pas de bonnes performances pour la vérification en milieu bruité comme c'est le cas dans les évaluation NIST SRE. Une alternative à l'application hasardeuse de réduction de vecteurs est l'utilisation de noyaux de séquences, qui permettent d'exprimer le critère d'apprentissage de manière adéquate à notre problème de classification de séquences, et ainsi de compacter judicieusement l'information structurée disponible en phase de développement. Notons qu'en mode "indépendant du texte" (sans information a priori sur le contenu prononcé dans les extraits à classer), les modèles stationnaires (MMG), qui considèrent les observations indépendants les unes des autres, donnent des performances similaires aux modèles dynamiques comme les modèles de Markov. Pour cela, nous nous limitons pour la vérification du locuteur aux noyaux entre ensembles de vecteurs, invariants à l'ordre chronologique des vecteurs [1], et que nous appelons dans la suite "noyau de séquences" par simplicité.

L'utilisation de noyaux de séquences pour la vérification du locuteur a été l'objet de plusieurs recherches ces dernières années. Dans [6, 12] par exemple, des noyaux de séquences basés sur des modèles génératifs ont été utilisés. Ces noyaux restent d'une complexité élevée dans le cas des MMG, communément utilisés pour capturer la distribution des paramètres acoustiques. Un noyau efficace qui a montré des performances prometteuses aux évaluations NIST SRE est le noyau GLDS [3]. Il consiste simplement en une projection explicite des séquences dans un espace de dimension fixe en utilisant une expansion polynomiale, suivie d'un produit scalaire linéaire.

Le noyau GLDS a cependant une limitation pratique et théorique. La première est que l'utilisation d'expansions polynomiales au delà d'un degré 3 n'est pas envisageable pour des problèmes de grande dimensionnalité. La seconde vient du fait qu'il n'est pas possible de généraliser l'approche en l'état à des expansions infinies (l'astuce du noyau n'est pas vraiment utilisée). Le but de cet article est d'aller au-delà de ces deux limitations. Nous commencons par définir une classe de noyaux de séquences dont le GLDS est un cas particulier, et développons une forme à dimension finie. Cette forme ayant une complexité élevée ("intractable") pour une application de vérification du locuteur, nous réduisons cette complexité grâce à une méthode de décomposition matricielle : la factorisation de Cholesky incomplète.

[1] Notons tout de même que l'information dynamique à court-terme est prise en compte dans les dérivées incluses dans ces vecteurs

2. GÉNÉRALISATION DU NOYAU GLDS

2.1. Présentation du noyau GLDS

La forme originale du noyau GLDS [3] fait intervenir une expansion polynomiale ϕ_p, composée de monômes jusqu'à un degré donné p. Par exemple, si $p = 2$ et $\mathrm{x} = [x_1, x_2]^\top$ est un vecteur à deux dimensions, $\phi_p(x) = [x_1, x_2, x_1^2, x_1x_2, x_2^2]^\top$. Le noyau entre deux séquences de vecteurs $X = \{\mathrm{x}_t\}_{t=1...T_X}$ et $Y = \{\mathrm{y}_t\}_{t=1...T_Y}$ est donné comme un produit scalaire normalisé entre expansions moyennes :

$$K(X,Y) = \tfrac{1}{T_X} \sum_{t=1}^{T_X} \phi_p(\mathrm{x}_t)^\top \, \mathbf{M}_p^{-1} \, \tfrac{1}{T_Y} \sum_{s=1}^{T_Y} \phi_p(\mathrm{y}_s) \quad (1)$$

où $\mathbf{M}_p$ est la matrice moment d'ordre 2 des expansions polynomiales ϕ_p estimée sur une population de développement, ou son approximation diagonale pour plus d'efficacité.

Conçu de cette manière, le noyau GLDS n'offre pas beaucoup de flexibilité pour coller aux données (peu de paramètres réglables), et l'expansion ϕ_p atteind des dimensions trop élevées pour un degré p supérieur à 3. Un problème intéressant est de trouver une manière d'implémenter (1) pour n'importe quel p, quitte à faire des approximations. Un problème plus général est d'aboutir à une forme finie de (1) pour n'importe quelle expansion ϕ, y compris les expansions infinies, de manière à pouvoir supporter des noyaux de types RBF. C'est le but de la prochaine section.

2.2. Une classe riche de noyaux

Nous considérons la classe de noyaux de séquences de la forme :

$$\begin{aligned} \hat{K}(X,Y) &= \tfrac{1}{T_X T_Y} \sum_{t=1}^{T_X} \sum_{s=1}^{T_Y} \phi(x_t)^\top \mathbf{M}^{-1} \phi(y_s) \\ &= \overline{\phi}(X)^\top \mathbf{M}^{-1} \overline{\phi}(Y) \end{aligned} \quad (2)$$

où :

- ϕ est une expansion vectorielle de taille $D \leq +\infty$ définissant un noyau de Mercer k :
$$k(\mathrm{x},\mathrm{y}) = \phi(\mathrm{x})^\top \phi(\mathrm{y}) \quad (3)$$
- $\overline{\phi}$ désigne la moyenne sur une séquence des expansions vectorielles ϕ (même notation dans la suite).
- $\mathbf{M} = E(\phi\phi^\top)$ est la matrice des moments d'ordre 2 des expansions ϕ estimée sur une population $B = \{\mathrm{b}_1, \ldots, \mathrm{b}_n\}$ de taille n. $\mathbf{M}$ peut être exprimée comme un produit matriciel faisant intervenir la matrice $(D \times n)$ des expansions de B, $\Phi_B = [\phi(\mathrm{b}_1), \ldots, \phi(\mathrm{b}_n)]$:
$$\mathbf{M} = \frac{1}{n} \Phi_B \Phi_B^\top \quad (4)$$

Remarquons que $\hat{k}(\mathrm{x},\mathrm{y}) = \phi(\mathrm{x})^\top \mathbf{M}^{-1} \phi(\mathrm{y})$ définit aussi un noyau satisfaisant la condition de Mercer, avec une normalisation dans l'espace caractéristique (*feature space*) défini par l'expansion ϕ. Le noyau de séquences peut être réécrit comme somme linéaire de ce noyau repondéré : $\hat{K}(X,Y) = \frac{1}{T_X T_Y} \sum_t \sum_s \hat{k}(\mathrm{x}_t, \mathrm{y}_s)$.

2.3. Expression de $\hat{K}$ dans la forme duale

Dans cette section, nous montrons comment exprimer le noyau vectoriel normalisé $\hat{k}$ en fonction du noyau vectoriel standard k défini en (3) et des données de normalisation $B = \{\mathrm{b}_1, \ldots, \mathrm{b}_n\}$.

Considérons la Décomposition en Valeur Singulière (SVD) mince [5] de la matrice des expansions Φ_B :

$$\Phi_B = USV^\top \quad (5)$$

où U and V sont des matrices orthogonales de tailles respectives $D \times r$ and $n \times r$, $r \leq \min(n, D)$ étant le rang de Φ_B. Nous pouvons décomposer de cette manière :

$$\mathbf{M} = \frac{1}{n} USV^\top \, VSU^\top = \frac{1}{n} US^2U^\top \quad (6)$$

Remarquons que dans le cas général, $\mathbf{M}$ n'est pas nécessairement inversible et doit être régularisée, en utilisant par exemple $\mathbf{M} = E(\phi\phi^\top) + \frac{1}{n}\varepsilon I$. Cette régularisation s'impose pour des raisons statistiques dans les cas où la dimension D est plus grande que le nombre de vecteurs n [11]. Toutefois, les approximations faites en section 3 permettent de se passer de cette régularisation. Nous nous limitons donc ici à une pseudo-inversion de (6) :

$$\mathbf{M}^{-1} = nUS^{-2}U^\top = n\Phi_B V S^{-4} V^\top \Phi_B^\top \quad (7)$$

La matrice de *Gram* sur B, définie par $\mathbf{K}_{i,j} = k(\mathrm{b}_i, \mathrm{b}_j)$, peut s'écrire $\mathbf{K} = \Phi_B^\top \Phi_B$. Selon (5), elle a une Décomposition en Valeur Singulière explicite $\mathbf{K} = VS^2V^\top$. En considérant la pseudo-inverse $\mathbf{K}^{-2} = VS^{-4}V^\top$, le noyau $\hat{k}$ peut s'écrire :

$$\begin{aligned} \hat{k}(x,y) &= n \;\; \phi(x)^\top \Phi_B \;\; \mathbf{K}^{-2} \;\; \Phi_B^\top \phi(y) \\ &= n \;\; \Psi_B(x)^\top \;\; \mathbf{K}^{-2} \;\; \Psi_B(y) \end{aligned} \quad (8)$$

où l'on définit la projection vectorielle de taille n via le noyau (3) :

$$\Psi_B(\mathrm{x}) = [k(\mathrm{b}_1, \mathrm{x}), \ldots, k(\mathrm{b}_n, \mathrm{x})]^\top \quad (9)$$

Par linéarité dans l'espace caractéristique, nous pouvons finalement écrire le noyau de séquences $\hat{K}$ dans une forme finie :

$$\hat{K}(X,Y) = n \, \overline{\Psi}_B(X)^\top \, \mathbf{K}^{-2} \, \overline{\Psi}_B(Y) \quad (10)$$

En pratique, le nombre de vecteurs disponibles pour le traitement de parole peut être très grand, et la taille n peut être énorme. L'implémentation de $\hat{K}$ en utilisant (10), avec une complexité en $O\left(n^2\right)$, peut donc vite être intractable. Dans la section suivante, nous utilisons une décomposition matricielle pour remédier à ce problème en donnant une forme approchée mais traitable de (10).

3. RÉDUCTION DE DIMENSIONNALITÉ

Les méthodes de réductions de données pour les méthodes à noyaux correspondent à des approximations de la matrice de Gram [4]. Le but de ces méthodes est de choisir un sous-ensemble $C \subset B$ qui permettrait d'approcher la matrice de Gram avec un rang inférieur, de manière à reformuler le problème avec une complexité moindre. Parmi ces techniques, la Factorisation de Cholesky Incomplète (ICD) [1] a une complexité relativement basse, en $O(m^2 n)$ si m est la taille désirée pour le sous-ensemble C. De plus, elle ne requiert pas le stockage en mémoire de l'intégralité de la matrice $\mathbf{K}$.

Etant donnée une matrice de Gram $\mathbf{K}$ de taille $n \times n$ (le rang de $\mathbf{K}$ pouvant être plus faible que n), l'ICD de $\mathbf{K}$ est une matrice $\mathbf{G}_m$ de taille $n \times m$ (de rang $m < n$),

telle que $\mathbf{K}$ paut être approchée par $\mathbf{G}_m{\mathbf{G}_m}^\top$. La racine $\mathbf{G}_m$ est générée par les colonnes de $\mathbf{K}$ indexées par $I = \{i_1, \ldots, i_m\} \subset \{1, \ldots, n\}$. Ainsi, l'on peut considérer que l'ICD fournit un *codebook* $C = \{\mathrm{b}_{i_1}, \ldots, \mathrm{b}_{i_m}\} \subset B$. Dans la suite, nous montrons comment exprimer notre noyau de séquences par une forme traitable utilisant C au lieu de B. Il peut être montré [1] que $\mathbf{G}_m$ peut s'écrire :

$$\mathbf{G}_m = \mathbf{K}(:, I)\mathbf{K}(I, I)^{-1/2} \tag{11}$$

où $\mathbf{K}(:, I)$ désigne toutes les colonnes de $\mathbf{K}$ indexées par I. Avec la même notation, $\mathbf{K}(I, I)$ est une matrice de Gram $m \times m$ sur les entrées $\{\mathrm{b}_{i_1}, \ldots, \mathrm{b}_{i_m}\}$.

Le fait que Φ_B et ${\mathbf{G}_m}^\top$ soient considérés comme ayant le même carré ($\mathbf{K} = \Phi_B^\top \Phi_B \approx \mathbf{G}_m{\mathbf{G}_m}^\top$) implique qu'il existe une matrice orthogonale U telle que l'on peut considérer la décomposition incomplète, à la place de (5) : $\Phi_B = U{\mathbf{G}_m}^\top$. La matrice $\mathbf{M}$ peut ainsi être approchée par $\frac{1}{n}U{\mathbf{G}_m}^\top\mathbf{G}_m U^\top$, ce qui revient à régulariser $\mathbf{M}$. Cette décomposition permet d'inverser :

$$\begin{aligned} \mathbf{M}^{-1} = &\quad n\, U\, ({\mathbf{G}_m}^\top\mathbf{G}_m)^{-1}\, U^\top \\ = &\quad U\ \mathbf{K}(I,I)^{1/2}\, \mathbf{R}^{-1}\, \mathbf{K}(I,I)^{1/2}\ U^\top \end{aligned} \tag{12}$$

où nous définissons selon (11) la matrice $m \times m$ (nécessairement inversible après l'ICD) :

$$\mathbf{R} = \frac{1}{n}\mathbf{K}(:, I)^\top \mathbf{K}(:, I) \tag{13}$$

Nous pouvons aussi déduire de (11) que $\Phi_B^\top = \mathbf{G}_m U^\top = \mathbf{K}(:, I)\mathbf{K}(I, I)^{-1/2}U^\top$. Si nous supposons que les expansions $\phi(\mathrm{x})$ appartiennent au sous-espace affine généré par les expansions incluses dans Φ_B, alors on peut généraliser

$$\phi(\mathrm{x})^\top = \Psi_C(\mathrm{x})^\top \mathbf{K}(I, I)^{-1/2}U^\top \tag{14}$$

où $\Psi_C(\mathrm{x})$ est la projection réduite sur les vecteurs de C dans l'espace caractéristique : $\Psi_C(\mathrm{x}) = [k(\mathrm{b}_{i_1}, \mathrm{x}), \ldots, k(\mathrm{b}_{i_m}, \mathrm{x})]^\top$. La formulation de notre nouveau noyau de séquences est finalement obtenue en injectant les nouvelles expressions de $\mathbf{M}^{-1}$ et $\phi(X)$ dans (2) :

$$\hat{K}_{ICDS}(X, Y) = \overline{\Psi}_C(X)^\top\, \mathbf{R}^{-1}\, \overline{\Psi}_C(Y) \tag{15}$$

La complexité de $\hat{K}_{ICDS}(X, Y)$ est en $O(m^2)$. En pratique, la valeur de m peut être choisie très inférieure à n, ce qui fait aboutir à une implémentation efficace d'un noyau.

Il est intéressant de remarquer que le noyau de séquences $\hat{K}_{ICDS}$ donné par (15) a une forme similaire à notre noyau de séquences RKHS défini dans notre précédent travail [8], où nous avions adopté la même stratégie que Campbell dans [3] pour concevoir un noyau mesurant la similarité entre deux séquences. Cette stratégie consiste à apprendre un modèle discriminant (avec pour valeurs cibles 0/1) sur une séquence (dans un Espace de Hilbert à Noyaux Reproduisants généré par k), et à l'appliquer sur une autre en supposant l'indépendance des observations. Après quelques approximations, un noyau symétrique vérifiant les conditions de Mercer est obtenu.

4. RÉSULTATS EXPÉRIMENTAUX

4.1. Corpus et pré-traitement

Nous testons notre système sur les données de NIST SRE 2004, en utilisant le protocole de développement défini par le projet Biosecure [2]. Dans ce protocole, nous considérons 113 locuteurs imposteurs pour développer le système. L'évaluation comprend plus de 7000 tests impliquant 181 locuteurs cibles et 368 séquences de test.

Les vecteurs acoustiques extraits des séquences de parole sont 12 MFCC auxquels nous rajoutons les dérivées temporelles premières. Un extracteur de silence utilisant un modèle bigaussien non-supervisé permet de rejeter les vecteurs correspondant aux segments de silence. La méthode de normalisation utilisée, après suppression des silences, est le *feature warping* [9].

4.2. Description du système

Une fois un noyau k choisi, la première étape est d'appliquer l'ICD à la matrice de gram estimée sur les données de développement. Dans les résultats montrés dans cet article, nous sélectionnons aléatoirement $n = 20000$ vecteurs (parmi les plus de 200000 disponibles) sur lesquels la décomposition est appliquée ; Des expérimentations ont montré que les performances étaient peu sensibles au nombre n de données de développement considérées du moment qu'il est suffisamment grand par rapport au nombre m de vecteurs retenus en sortie de l'ICD. Pour un bon compromis entre complexité et performance, nous avons choisi de retenir $m \sim 5000$ vecteurs *codebook*. Une fois la projection $\overline{\Psi}_C$ déterminée par le choix de k et du *codebook* C, la matrice de normalisation $\mathbf{R}$ définie par (13) peut être calculée.

Les modèles SVM des locuteurs cibles sont entrainés en considérant un ensemble commun de séquences imposteurs, dont les caractéristiques (valeurs du noyau (15) entre toutes les paires de séquences) peuvent être calculées à l'avance et stockées en mémoire. Pour rendre plus efficace l'apprentissage des modèles, les projections des séquences imposteurs sont mémorisées sous la forme $\mathbf{R}^{-1}\overline{\Psi}_C$. De cette manière, quand une séquence S émise par un locuteur cible est donnée au système, il n'y a qu'à calculer $\overline{\Psi}_C(S)$ pour obtenir les valeurs du noyau avec les séquences imposteurs par un simple produit scalaire.

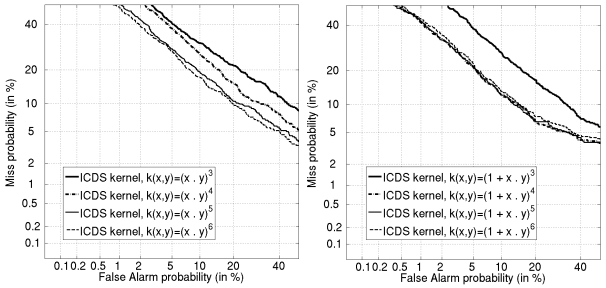
La procédure de test peut être rendue efficace par une astuce similaire. La fonction discriminante pour un locuteur peut être compactée dans un vecteur de dimension m (de manière analogue à [8]).

4.3. Choix des paramètres du noyau

Dans cette section, nous discutons comment choisir les paramètres du noyau k choisi pour définir $\hat{K}$.

Noyaux polynomiaux de la forme $k_p(\mathrm{x}, \mathrm{y}) = (c + \mathrm{x}\cdot\mathrm{y})^p$ Les résultats correspondants aux valeurs $c = 0$ (Fig.1.a) et $c = 1$ (Fig.1.b) montrent qu'il vaut mieux prendre un valeur non nulle pour c. Il est donc préférable de tenir compte de tous les monômes avec un degré inférieur ou égal à p, comme avec le noyau GLDS (quand $c = 0$, seulement les monômes de degrés p sont pris en compte).

De plus, les Figures 1.a et 1.b montrent que les performances sont meilleures avec un degré plus grand que 3, ce qui suggère que le noyau GLDS donnerait de meilleures performances en considérant aussi les monômes d'un degré supérieur à 3 dans l'expansion ϕ_p.



(a) forme $(x \cdot y)^p$ (b) forme $(1 + x \cdot y)^p$

FIG. 1: Performances des noyaux polynomiaux

Noyaux RBF de la forme $k_{rbf}(\mathrm{x},\mathrm{y}) = e^{-\gamma|\mathrm{x}-\mathrm{y}|^2}$ [11] recommande de choisir γ de l'ordre de $\gamma_0 = 1/2d\sigma^2$, où σ est la moyenne des écarts-types pour les composantes des vecteurs d'observation. Avec notre pré-traitement, ceci correspond à $\gamma_0 \approx 0.3$. Nos expériences confortent ce choix (fig.2). En fait, si γ est trop élevé, le noyau vectoriel colle trop aux données, $\mathbf{K}$ est proche de la matrice identité (rang maximal), et la projection de séquence définie en (15) revient à compter combien de vecteurs de la séquence gisent dans le voisinage de chaque vecteur *codebook*. Au contraire, si γ est trop faible, le rang de $\mathbf{K}$ est faible, et un nombre trop faible de caractéristiques sera considéré.

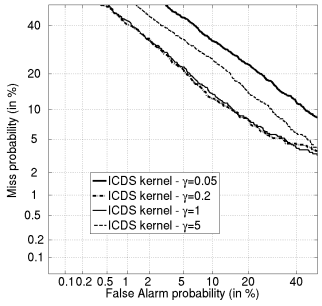


FIG. 2: Performances pour les noyaux RBF

4.4. Comparaison avec d'autres systèmes SVM

Les meilleures performances de notre noyau ICDS sont obtenus avec un noyau RBF (les résultats sont montrés avec $\gamma = 0.2$). Dans fig.3, elles sont comparées à celles d'autres noyaux de séquence (modélisation SVM) :

- L'approche GLDS [3] avec une expansion polynomiale de degré 3 et une approximation diagonale de la matrice second moment $\mathbf{M}_p$.
- La même approche avec une matrice pleine $\mathbf{M}_p$.
- Notre approche précédente [8] (noyau RKHSS).

Les résultats montrent que notre système donne de meilleures performances que les autres. Des expérimentations avec d'autre paramétrisation (exploitant l'information spectrale) confirment cette tendance.

5. CONCLUSION

Nous avons présenté une manière de généraliser le noyau GLDS à un espace caractéristique défini par un noyau vectoriel de Mercer quelconque. Le noyau de séquences ainsi conçu a été rendu implémentable en utilisant une décomposition de la matrice de gram correspondant au noyau vectoriel. Il conduit à des performances meilleures que le

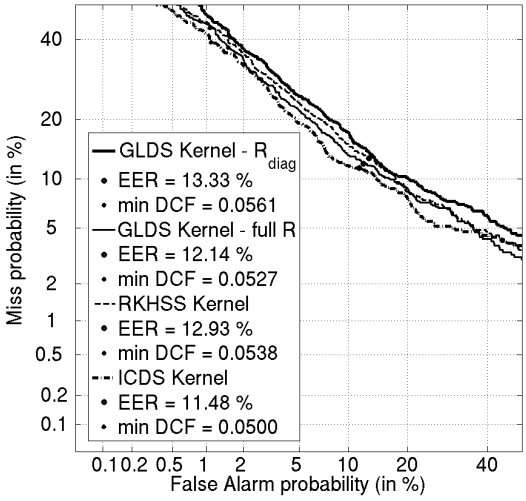


FIG. 3: Comparaison de plusieurs systèmes SVM

GLDS et que notre ancienne approche similaire. Plusieurs extensions sont possibles. Par exemple, il serait intéressant de considérer la matrice de covariance à la place de la matrice de second moment $\mathbf{M}$, afin de définir un noyau basé sur la distance de Malahanobis dans l'espace caractéristique.

RÉFÉRENCES

[1] F.R. Bach and M.I. Jordan. Predictive low-rank decomposition for kernel methods. In *Proc. ICML*, 2005.

[2] Biosecure network of excellence : Biometrics for secure authentification. http ://www.biosecure.info, 2005.

[3] W.M. Campbell, J. Campbell, D. Reynolds, E. Singer, and P. Torres-Carrasquillo. Support vector machines for speaker and language recognition. *Computer Speech and Language*, 2005.

[4] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2, 2001.

[5] G.H. Golub and C.F. Van Loan. *Matrix Computation*. The John Hopkins Univ. Press, 1996.

[6] P. Ho and P Moreno. SVM kernel adaptation in speaker classification and verification. In *Proc. ICSLP*, 2004.

[7] Z. Lei, Y. Yang, and Z. Wu. Mixture of support vector machines for text-independent speaker recognition. In *Proc. Interspeech*, 2005.

[8] J. Louradour and K. Daoudi. Conceiving a new sequence kernel and applying it to SVM speaker verification . In *Proc. Interspeech*, 2005.

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

[10] J. Platt. *Probabilities for SV Machines*. MIT Press, 2000.

[11] B. Schölkopf, S. Mika, C. J.C.Burges, P. Knirsch, K.-R. Müller, G. Rätsch, and A. J. Smola. Input space versus feature space. *IEEE Trans. Neural Networks*, 100(5) :1000–1017, 1999.

[12] V. Wan. *Speaker Verification using Support Vector Machines*. PhD thesis, University of Sheffield, 2003.