ExpLSA et classification de textes

Nicolas Béchet, Mathieu Roche, Jacques Chauché
Équipe TAL, LIRMM - UMR 5506, CNRS
Université Montpellier 2, 34392 Montpellier Cedex 5 - France

Abstract

Latent semantic analysis (LSA) is a statistic method which can be used to perform the classification of texts approaches. The size of the textual data has usefully an important impact for this task. This paper proposes an expansion method of sentences to improve the classification approaches. Experimental results based on a corpus of news allow to characterize the texts which improve LSA.

Résumé

L'analyse sémantique latente (LSA - Latent Semantic Analysis) est une méthode statistique qui peut être utilisée pour des tâches de classification de textes. La quantité des données textuelles (taille des documents à classer) est souvent un critère déterminant pour de telles tâches. Ainsi, cet article propose une méthode d'expansion des phrases des textes afin d'enrichir les données textuelles et améliorer les méthodes de classification. Les résultats expérimentaux obtenus à partir d'un corpus de dépêches d'actualité ont permis de caractériser les types de textes pour lesquels notre méthode améliore LSA.

Mots-clés : Classification de textes, LSA, analyse syntaxique.

1. Introduction

Nous présentons dans cet article des méthodes de classification automatique de documents textuels (catégorisation thématique). Pour une telle tâche, la première étape que nous avons mise en œuvre consiste à appliquer une méthode appelée Latent Semantic Analysis (LSA) développée par (Landauer et Dumais, 1997)[1]. La méthode LSA est uniquement fondée sur une approche statistique appliquée à des corpus de grande dimension. L'approche LSA consiste à représenter les données textuelles dans le but de regrouper les documents thématiquement proches. Pour ce faire, un espace sémantique associant chaque document à un vecteur est retourné. Un calcul de similarité entre deux vecteurs (comme le cosinus) permet alors d'évaluer la proximité de deux documents. L'objectif de nos travaux est d'améliorer les performances de LSA par une approche nommée *ExpLSA* (***Exp****ansion des contextes avec* ***LSA***) dans un contexte de classification de textes. L'approche *ExpLSA* consiste à enrichir le corpus pour finalement appliquer une analyse sémantique latente *classique*. Nous cherchons avec cette méthode à combler la faible quantité de descripteurs contenus dans les documents (ou contextes) en ajoutant de l'information.

Les méthodes LSA et *ExpLSA* constituent une première étape de représentation des données textuelles dans le but d'appliquer des méthodes de classification automatique utilisées en fouille de données. Nous allons en effet évaluer notre méthode *ExpLSA* en utilisant divers

[1] voir aussi, http://www.msci.memphis.edu/~wiemerhp/trg/lsa-followup.html

algorithmes de classification automatique : les KPPV (K Plus Proches Voisins), l'approche bayésienne naïve, les SVM (machines à support vectoriel) et C4.5. Ces méthodes seront rigoureusement décrites dans la suite de cet article. Des expérimentations menées à partir d'un corpus de dépêches journalistiques provenant du site web français Yahoo (http://fr.news.yahoo.com/) nous permettent d'évaluer les résultats obtenus en utilisant l'approche *ExpLSA*. Cette méthode est détaillée dans (Béchet et al., 2008), son principe global sera résumé par la suite.

La section suivante propose un résumé des caractéristiques théoriques et des limites de LSA. La section 3 établie un état de l'art dans le domaine de l'utilisation de connaissances syntaxiques associées à LSA. Nous présentons ensuite notre méthode *ExpLSA* (section 4). La section 5 montre l'intérêt d'un enrichissement afin d'effectuer une classification de textes. Le protocole expérimental utilisé sera décrit dans la section 6 pour finalement présenter les résultats obtenus.

2. LSA

La méthode LSA qui s'appuie sur l'hypothèse "harrissienne", est fondée sur le fait que des mots qui apparaissent dans un même contexte sont sémantiquement proches. Le corpus est représenté sous forme matricielle. Les lignes sont relatives aux mots et les colonnes représentent les différents contextes choisis (un document, un paragraphe, une phrase, etc.). Chaque cellule de la matrice représente le nombre d'occurrences des mots dans chacun des contextes du corpus. Deux mots proches au niveau sémantique sont représentés par des vecteurs proches. La mesure de proximité est généralement définie par le cosinus de l'angle entre les deux vecteurs.

2.1. Caractéristiques théoriques de LSA

La théorie sur laquelle s'appuie LSA est la décomposition en valeurs singulières (SVD). Une matrice $A = [a_{ij}]$ où a_{ij} est la fréquence d'apparition du mot i dans le contexte j, se décompose en un produit de trois matrices USV^T. U et V sont des matrices orthogonales et S une matrice diagonale.

Soit S_k où $k < r$ la matrice produite en enlevant de S les $r - k$ colonnes qui ont les plus petites valeurs singulières. Soit U_k et V_k les matrices obtenues en enlevant les colonnes correspondantes des matrices U et V . La matrice $U_kS_kV_k^T$ peut alors être considérée comme une version compressée de la matrice originale A. Les expériences décrites dans la section 6 ont été menées avec un nombre de facteurs k égal à 100.

Nous précisons qu'avant d'effectuer la décomposition en valeurs singulières, une première étape de normalisation de la matrice d'origine A est exécutée. Cette normalisation consiste à appliquer un logarithme et un calcul d'entropie sur la matrice A. Ainsi, plutôt que de se fonder directement sur le nombre d'occurrences de chacun des mots, une telle transformation permet de s'appuyer sur une estimation de l'importance de chacun des mots dans leur contexte. De manière similaire aux travaux de (Turney, 2001), cette étape de normalisation peut également s'appuyer sur la méthode du $tf \times idf$, approche bien connue dans le domaine de la Recherche d'Information.

Précisons de plus que nous ne prenons pas en compte les ponctuations ainsi qu'un certain nombre de mots non significatifs du point de vue sémantique tels que les mots "et", "à", "le", etc. (mots outils).

2.2. Les limites de LSA

LSA offre des avantages parmi lesquels, la notion d'indépendance par rapport à la langue du corpus étudié, le fait de se dispenser de connaissances linguistiques ainsi que de celles du domaine tels que des thésaurus. Bien que cette approche soit pertinente pour les tâches de classification, il n'en demeure pas moins que son utilisation soulève des contraintes.

Notons tout d'abord l'importance de la taille des contextes choisis. (Rehder et al., 1998) ont montré lors de leurs expérimentations que si les contextes possèdent moins de 60 mots, les résultats s'avèrent être décevants.

Il a également été mis en évidence par (Roche et Chauché, 2006) que l'efficacité de LSA est influencée par la proximité du vocabulaire utilisé. En effet, l'homogénéité des corpus sur le plan thématique donne des résultats décevants avec LSA.

Pour résoudre de tels problèmes, une des solutions peut consister à ajouter des connaissances syntaxiques à LSA, comme cela est décrit dans la section suivante.

3. État de l'art sur l'ajout de connaissances syntaxiques à LSA

(Landauer et al., 1997) posent le problème du manque d'informations syntaxiques dans LSA en comparant cette méthode à une évaluation humaine. Il est question de proposer à des experts humains d'attribuer des notes à des essais sur le cœur humain de 250 mots rédigés par des étudiants. Un espace sémantique a été créé à partir de 27 articles écrits en anglais traitant du cœur humain "appris" par LSA. Les tests effectués concluent que la méthode LSA obtient des résultats satisfaisants comparativement à l'expertise humaine. Il en ressort que les mauvais résultats étaient dus à une absence de connaissances syntaxiques dans l'approche utilisée. Ainsi, les travaux qui sont décrits ci-dessous montrent de quelle manière de telles connaissances peuvent être ajoutées à LSA.

La première approche de (Wiemer-Hastings et Zipitria, 2001) utilise des étiquettes grammaticales Brill, 1994 appliquées à l'ensemble du corpus étudié (corpus de textes d'étudiants). Les étiquettes étant rattachées à chaque mot avec un blanc souligné ("_"), l'analyse qui s'en suit via LSA considère le mot associé à son étiquette comme un seul terme. Les résultats de calculs de similarités obtenus avec une telle méthode restent décevants. Notons que de telles informations grammaticales ne sont pas des connaissances syntaxiques proprement dites contrairement à la seconde approche de (Wiemer-Hastings et Zipitria, 2001) décrite ci-dessous. Cette seconde approche se traduit par l'utilisation d'un analyseur syntaxique afin de segmenter le texte avant d'appliquer l'analyse sémantique latente. Cette approche est appelée "LSA structurée" (SLSA). Une décomposition syntaxique des phrases en différents composants (sujet, verbe, objet) est tout d'abord effectuée. La similarité est ensuite calculée en traitant séparément par LSA les trois ensembles décrits précédemment. Les similarités (calcul du cosinus) entre les vecteurs des trois matrices formées sont alors évaluées. La moyenne des similarités est enfin calculée. Cette méthode a donné des résultats satisfaisants par rapport à "LSA classique" en augmentant la corrélation des scores obtenus avec les experts pour une tâche d'évaluation de réponses données par des étudiants à un test d'informatique.

(Kanejiya et al., 2003) proposent un modèle appelé SELSA. Au lieu de générer une matrice de co-occurrences mot/document, il est proposé une matrice dans laquelle chaque ligne contient toutes les combinaisons mot/étiquette et en colonne les documents. L'étiquette "préfixe" renseigne sur le type grammatical du voisinage du mot traité. Le sens d'un mot est

en effet donné par le voisinage grammatical duquel il est issu. Cette approche est assez similaire à l'utilisation des étiquettes de (Brill, 1994) présentée dans les travaux de (Wiemer-Hastings et Zipitria, 2001). Mais SELSA étend ce travail vers un cadre plus général où un mot avec un contexte syntaxique spécifié par ses mots adjacents est considéré comme une unité de représentation de connaissances.

L'évaluation de cette approche a montré que la méthode LSA était plus pertinente que SELSA dans un test de corrélation avec des experts. Cependant, SELSA se révèle plus précise pour ce qui est de tester les bonnes et mauvaises réponses (SELSA fait moins de fautes que LSA mais en retourne de plus nuisibles).

L'approche *ExpLSA* que nous présentons dans cet article se place dans un contexte différent. En effet, nous proposons d'utiliser la régularité de certaines relations syntaxiques afin d'enrichir le contexte (document).

L'utilisation de ressources lexicales et sémantiques pour enrichir des contextes est un concept répandu dans le domaine de la recherche d'informations textuelles, pour des tâches d'indexation ou d'expansion de requêtes. La plupart de ces approches utilisent des ressources lexicales génériques (Voorhees, 1994 ; Moldovan et Mihalcea, 2000) en ajoutant des termes reliés sémantiquement aux termes d'origine.

Notre approche propose quant à elle, pour une tâche de classification de textes, d'utiliser les ressources du corpus étudié afin d'enrichir le document. Une telle approche ne pourrait pas être utilisée pour une tâche d'expansion de requêtes car le contexte (mots clés) est trop pauvre et ne permettrait pas d'extraire suffisamment d'informations. De plus notre approche présentée dans la section suivante nécessite une analyse syntaxique à partir de phrases syntaxiquement bien formées, ce qui est impossible avec ce type de contexte « mots clés ».

4. Notre approche : *ExpLSA*

L'approche *ExpLSA* propose d'enrichir un corpus lemmatisé en effectuant une expansion des phrases. Cette expansion se fonde sur une méthode syntaxique afin de compléter les mots du corpus avec des mots jugés sémantiquement proches. Notre méthode est décrite de manière précise dans (Béchet et al., 2008). Les sections suivantes résument le principe de notre approche qui sera appliquée pour une tâche de classification de textes.

4.1. Utilisation d'un analyseur syntaxique

Nous utilisons dans un premier temps l'analyseur syntaxique SYGFRAN (Chauché, 1984) afin d'extraire du corpus les relations syntaxiques Verbe-Objet (Verbe_Préposition_Complément, Verbe_COD).

Nous avons par exemple extrait de la phrase « *L'accompagnement nécessite des professionnels.* » la relation syntaxique « *verbe : nécessiter, COD : professionnels* ».

Une fois la totalité des relations syntaxiques extraites, nous lemmatisons le corpus en utilisant le système SYGMART (Chauché, 1984).

4.2. Regroupement des objets en fonction de la proximité des verbes

Nous évaluons ensuite la proximité sémantique entre les verbes. Pour cela, nous utilisons la mesure d'Asium (Faure, 2000). Cette mesure considère deux verbes comme proches s'ils possèdent un nombre important d'objets en commun en fonction du nombre total d'objets de chaque verbe.

Après avoir étudié la proximité sémantique entre les verbes en évaluant chaque verbe du corpus avec tous les autres, nous ne conservons que le couple de verbes ayant obtenu le meilleur score de similarité avec la mesure d'Asium. Citons par exemple les verbes *requérir* et *nécessiter* qui partagent communément les objets *professionnel*, *courage* et *connaissance*.

Ainsi, nous regroupons tous les objets communs dont les verbes ont été jugés proches sémantiquement par le seuil de similarité le plus élevé parmi l'ensemble des couples de verbes. Nous complétons alors le corpus initial en attachant à chaque mot les objets communs déterminés par la mesure d'Asium[2]. L'exemple ci-dessous illustre la méthode appliquée.

La phrase :

- *L'accompagnement nécessite des professionnels.*

qui une fois lemmatisée :

- *L'accompagnement nécessiter de le professionnel.*

va être enrichie avec la méthode d'expansion :

- *L'accompagnement nécessiter de le (professionnel courage connaissance).*

La dernière étape d'*ExpLSA* est l'application de l'approche LSA sur le corpus enrichi.

5. L'enrichissement appliqué à la classification de textes

La classification de textes consiste à regrouper des contextes (dans notre cas des documents) dans différentes classes qui correspondent à des catégories thématiques (par exemple, les thèmes "*politique*", "*sport*", "*technologies*", etc).

Un contexte contient des descripteurs qui peuvent être insuffisants pour la réalisation d'une classification automatique. Notre approche *ExpLSA* propose une solution à ce manque d'information en enrichissant le contexte. Prenons par exemple les phrases suivantes :

• P1 : *Le député s'adresse aux consuls*

• P2 : *Le sénateur s'exprime devant les ambassadeurs*

On constate que les phrases P1 et P2 n'ont aucun mot en commun (sans considérer les mots outils) ce qui classerait ces phrases dans deux catégories différentes en nous appuyant sur des méthodes statistiques. Après expansion avec la méthode décrite dans les sections précédentes, il est possible d'enrichir ces phrases de la manière suivante (pour faciliter la lecture de cet exemple, la lemmatisation n'a pas été ici répercutée) :

• E1 : *Le (député sénateur parlementaire) s'adresse aux (consuls ambassadeurs diplomates)*

• E2 : *Le (sénateur député parlementaire) s'exprime devant les (ambassadeurs consuls diplomates)*

Dans ce cas, les phrases E1 et E2 possèdent six mots communs signifiant une proximité thématique. On montre par cet exemple que deux phrases proches sémantiquement peuvent s'avérer difficiles à classer sans utiliser de connaissances sémantiques. Avec notre

[2]Différentes méthodes d'enrichissement ont été expérimentées dans (Béchet et al., 2008) pour une tâche de classification conceptuelle. Nous utiliserons la méthode d'expansion qui a donné les résultats les plus significatifs. Celle-ci consiste à enrichir les phrases avec les objets communs des verbes.

enrichissement, l'information apportée peut remédier à cette problématique. Rappelons que nous plaçons nos travaux dans le cadre du traitement de textes qui peuvent se révéler plus ou moins spécialisés sans utiliser de connaissances sémantiques tels que des dictionnaires du domaine. Ainsi, notre approche est indépendante du domaine car l'enrichissement effectué s'appuie sur les données mêmes du corpus.

Après avoir succinctement présenté le contexte de l'application de *ExpLSA* dans le cadre de la classification de textes, nous présentons les expérimentations menées pour une telle tâche.

6. Expérimentations

Pour discuter de la qualité des résultats retournés avec notre approche, nous nous appuyons sur le protocole expérimental décrit dans la section suivante.

6.1. Protocole expérimental

Pour mener ces expérimentations, nous nous référons à un corpus constitué d'un ensemble de dépêches d'actualité en français issues du site de Yahoo (http://fr.news.yahoo.com/). Il se compose de 2828 articles et est constitué de 914540 mots (5,3 Mo). Il est divisé en onze classes : *france*, *économie*, *insolite*, *santé*, *monde*, *politique*, *culture*, *sciences*, *people*, *technologies*, *sport*.

En effet, l'objectif des expérimentations est de comparer la méthode LSA à notre approche *ExpLSA* en réalisant une classification automatique des articles dans un contexte de classification supervisée. Pour cela, nous avons expérimenté quatre algorithmes de classification supervisée[3]. Nous présentons ci-dessous une description succincte de ces approches qui sont détaillées dans (Cornuéjols et Miclet, 2002). Ainsi, les entrées de ces algorithmes issus du domaine de la fouille de données seront les vecteurs retournés par LSA et *ExpLSA*.

• Les **K Plus Proches Voisins (KPPV)**. L'algorithme des KPPV détermine la classe d'un nouveau texte en lui attribuant la classe majoritaire des K textes les plus proches, en terme de mesure de proximité, issue de la base d'apprentissage. Dans notre cas, la mesure de similarité utilisée entre deux vecteurs représentant un texte est le cosinus. Notons qu'à chaque nouvelle classification, il est nécessaire de parcourir l'ensemble de la base d'apprentissage. Cet algorithme est donc coûteux en terme de temps mais offre, en général, des résultats parfaitement satisfaisants.

• L'**approche bayésienne naïve**. Lors d'une classification bayésienne naïve, une classe est déterminée de la manière suivante. Pour un ensemble de classes possibles C et une instance spécifiée par un ensemble d'attributs A, la valeur de classification bayésienne naïve c est définie comme suit :

$$c = argmax P(c_j)_{c_j \in C} \prod_{a_i \in A} P(a_i|c_j) \text{ avec } P(a_i|c_j) = \frac{P(c_j|a_i) \times P(a_i)}{P(c_j)} \text{ où :}$$

- $P(a_i)$ est la probabilité que l'hypothèse a_i soit vérifiée indépendamment des données c_j.
- $P(a_i|c_j)$ est la probabilité d'observer les données a_i sachant que l'hypothèse c_j est vérifiée.

[3]Nous utilisons l'outil Weka afin d'appliquer ces algorithmes. http : //www.cs.waikato.ac.nz/~ml/

Cet algorithme, détaillé dans (Cornuéjols et Miclet, 2002), présente la particularité d'être très rapide et est un bon compromis vitesse-qualité.

• Les **SVM (Support Vector Machine - Machines à support vectoriel)**. Le cas d'une machine à support vectoriel consiste à trouver un hyperplan séparateur entre les classes. La classe d'appartenance est déterminée en utilisant une fonction de décision. Celle-ci est calculée à partir des points appartenant à la frontière de l'estimation du support, c'est-à-dire pour les points se situant précisément sur l'hyperplan : les vecteurs support. Nous utilisons dans nos expérimentations l'algorithme SMO (Platt, 1999), optimisant le problème dual des SVM, permettant une utilisation des SVM pour un problème multi-classes. L'application de cet algorithme est plus lent que l'utilisation d'un classificateur bayésien, mais offre globalement de meilleures performances.

• Une méthode de construction d'**arbre de décision**, **C4.5**. Un arbre de décision est un arbre tel qu'un nœud correspond à un attribut, les feuilles issues de ce nœud ont des valeurs possibles pour cet attribut. Les feuilles correspondent à une classe. Le chemin parcouru par un nouvel exemple de la racine de l'arbre jusqu'à la feuille détermine sa classe. La principale difficulté est liée à l'élaboration d'un tel arbre afin de construire une base d'apprentissage. C4.5 est un algorithme permettant cette construction. Cet algorithme vise à séparer en un ensemble le plus homogène possible des cas exemples. Cette séparation s'appuie sur l'entropie (Cornuéjols et Miclet, 2002) qui mesure la quantité d'information.

Afin d'estimer la fiabilité de ces algorithmes, nous appliquons un processus de validation croisée[4] en segmentant les données en dix sous-ensembles. Cette méthode permet en effet de considérer alternativement les sous-ensembles comme des ensembles d'apprentissage ou comme des jeux de test. Ces derniers sont évalués en utilisant les mesures de rappel et de précision pour chaque classe *i* qui sont définies comme suit :

$$\text{précision}_i = \frac{\text{nombre d'articles correctement attribués à la classe } i}{\text{nombre d'article attribués à la classe } i} \quad (1)$$

$$\text{rappel}_i = \frac{\text{nombre d'articles correctement attribués à la classe } i}{\text{nombre d'articles appartenant à la classe } i} \quad (2)$$

En général, il est important de déterminer un compromis entre le rappel et la précision. Pour cela, nous pouvons utiliser une mesure prenant en compte ces deux critères d'évaluation en calculant le f*score*$_i$ pour chaque classe *i* et le Fscore moyen :

$$fscore_i(\beta) = \frac{(\beta^2+1) \times p_i \times r_i}{\beta^2 \times p_i + r_i} \text{ avec } r_i = \text{rappel}_i \text{ et } p_i = \text{précision}_i \quad (3)$$

$$Fscore(\beta) = \frac{\sum_{i=1}^{k} fscore_i(\beta)}{k} \text{ avec } k \text{ le nombre total de classes} \quad (4)$$

Le paramètre β des formules (3) et (4) permet de régler les influences respectives de la précision et du rappel. Il est très souvent fixé à 1 pour accorder le même poids à ces deux mesures d'évaluation. Dans les sections suivantes, nous nous appuierons sur la mesure de *Fscore* avec $\beta = 1$.

[4]Réalisée avec l'application Weka.

6.2. Résultats

Nous proposons dans un premier temps de comparer LSA et *ExpLSA* en utilisant les quatre classificateurs sur la totalité du corpus en appliquant une validation croisée (10-CV). Des expérimentations avec deux seuils SA très différents pour la mesure d'Asium sont présentées dans la suite de cet article : SA = 0.6 et SA = 0.9. De ce fait, nous appliquons un enrichissement important mais relativement bruité avec un seuil SA faible (0.6) et un faible enrichissement mais de meilleure qualité avec un seuil SA élevé (0.9).

Classificateur	*Méthode*	*Fscore Moyen*
KPPV	LSA	**59,45%**
	ExpLSA 0,6	43,84%
	ExpLSA 0,9	57,36%
NaiveBayes	LSA	**65,54%**
	ExpLSA 0,6	55,13%
	ExpLSA 0,9	64,84%
SVM	LSA	**66,68%**
	ExpLSA 0,6	56,84%
	ExpLSA 0,9	65,23%
C4.5	LSA	**52,79%**
	ExpLSA 0,6	38,15%
	ExpLSA 0,9	51,35%

Table 1: Comparaison de LSA et ExpLSA (seuils de 0.6 et 0.9)

Le tableau 1 montre les résultats de cette première expérimentation en évaluant le *Fscore* moyen des classes suivant les différentes méthodes. Notons qu'avec un seuil SA de 0.6, la méthode *ExpLSA* dégrade fortement les résultats ce qui confirme les résultats préliminaires présentés dans (Béchet et al., 2007). Avec un seuil SA de 0.9, les performances des algorithmes sont légèrement dégradées de l'ordre de 1 à 2 %. D'autres expérimentations ont montré qu'avec un seuil plus faible que 0.9, le *Fscore* moyen est significativement plus faible que LSA. Ainsi, pour cette tâche de classification de textes, le seuil SA = 0.9 sera retenu.

Le tableau 1 montre également que le classificateur *SVM* obtient les meilleurs résultats avec le *Fscore* quelque soit la méthode (LSA ou *ExpLSA*). La tableau 3 en annexe montre l'ensemble des résultats obtenus avec les méthodes LSA et *ExpLSA* pour les différentes classes utilisant le classificateur *SVM*, ainsi que les matrices de confusion correspondantes.

Ainsi, même si les résultats de *ExpLSA* sont plus faibles en terme de *Fscore* moyen comparativement à LSA, nous proposons de caractériser les types de textes plus ou moins adaptés à chaque méthode. Les expérimentations présentées ci-dessous (avec SA = 0.9) ont alors consisté à réitérer les mêmes évaluations en considérant la taille des articles. En effet, l'ensemble des articles a été répertorié en trois tailles :

• articles courts comptant moins de 250 mots,

• articles de taille moyenne de 251 à 500 mots,

• articles longs composés de plus de 500 mots.

Ces seuils propres aux tailles ont été déterminés expérimentalement afin d'obtenir un nombre de textes du même ordre pour chaque catégorie de taille.

		Taille des articles		
		court	moyen	long
Classificateu	*Méthode*	*Fscore Moyen*	*Fscore Moyen*	*Fscore Moyen*
KPPV	LSA	**49,96%**	55,71%	57,09%
	ExpLSA 0,9	48,06%	**56,08%**	**58,45%**
NaiveBayes	LSA	61,13%	**62,94%**	**64,08%**
	ExpLSA 0,9	**63,64%**	61,71%	63,64%
SVM	LSA	**58,58%**	61,27%	62,03%
	ExpLSA 0,9	57,32%	**61,55%**	**62,18%**
C4.5	LSA	**49,55%**	44,79%	48,96%
	ExpLSA 0,9	43,83%	**49,25%**	**50,43%**

Table 2: Comparaison de LSA et ExpLSA *en fonction de la taille des articles*

Le tableau 2 présente le même calcul du *Fscore* que dans le tableau 1 pour les trois types d'articles triés suivant leur taille. En considérant les classificateurs *SVM*, *KPPV* et *C4.5*, les résultats pour les articles de taille moyenne et importante sont améliorés avec *ExpLSA*. *ExpLSA* étend le contexte avec des mots pertinents mais l'enrichissement peut également ajouter du bruit. Or, si l'enrichissement apporte partiellement du bruit, il sera réduit par l'approximation de la matrice obtenue par la décomposition en valeurs singulières propre à LSA appliquée aux textes de taille importante. Mais à l'inverse, l'application de LSA sur un article court, contenant donc peu de descripteurs, peut accroître le bruit généré par l'enrichissement. Cette situation peut expliquer que, globalement, les résultats de classification sont améliorés pour les textes de taille moyenne et importante contrairement aux documents de taille réduite. Le classificateur bayésien naïf propose quant à lui des résultats opposés. Nous pouvons supposer que le bon comportement de ce classificateur pour les articles courts peut être la conséquence de sa robustesse aux données partielles (Fitzgerald, 1999). Ces résultats feront l'objet d'une étude plus approfondie afin d'expliquer précisément ces résultats.

Notons que les valeurs du tableau 2 sont plus faibles que ceux du tableau 1 car les données d'apprentissage sont moins importantes dans le premier cas. Nous utilisons en effet dans la seconde expérimentation des sous-ensembles du corpus : articles courts, moyens et longs. Ceci permet d'expliquer l'écart obtenu dans les résultats des deux tableaux. Remarquons également que l'algorithme bayésien naïf obtient de meilleures performances que les *SVM* dans le tableau 2 par rapport au tableau 1. Cela nous permet d'émettre l'hypothèse que cet algorithme se comporte mieux que *SVM* avec un corpus d'apprentissage plus faible.

Les résultats présentés dans cette section nous permettent de proposer une méthode mixte LSA/*ExpLSA* afin d'améliorer de manière globale les méthodes de classification. Avec cette approche mixte, avant de classifier un texte, nous proposons d'évaluer sa taille. Si cette dernière est faible, l'application de LSA est effectuée, sinon l'approche *ExpLSA* est retenue. Des futurs travaux devront déterminer de manière plus précise les différents seuils de taille des textes.

7. Conclusion et discussions

LSA est une méthode statistique utilisée entre autres pour regrouper des contextes afin d'établir une classification de textes. Néanmoins, cette méthode donne des résultats parfois décevants. Ceux-ci s'expliquent notamment par l'absence de connaissances linguistiques.

Nous avons proposé dans cet article l'approche *ExpLSA*. Celle-ci consiste à effectuer une expansion des contextes avant d'appliquer LSA. Nous rendons de ce fait les contextes plus

riches en utilisant des outils syntaxiques afin d'y parvenir. Les approches LSA et *ExpLSA* constituent une première étape lors d'une classification de textes.

Nous avons en effet utilisé pour évaluer notre méthode quatre classificateurs : les KPPV (K Plus Proches Voisins), l'approche bayésienne naïve, les SVM (machines à support vectoriel) et C4.5. Nos premières expérimentations prenant en compte la totalité du corpus ont montré que notre approche réduisait les performances de LSA (sur la base du *Fscore* moyen). Cependant, en considérant la taille des textes, nos expérimentations ont permis de mettre en relief que *ExpLSA* améliorait les résultats de LSA avec les articles de taille moyenne et importante pour les classificateurs SVM, KPPV et C4.5. Nous envisageons de mener de nouvelles expérimentations avec des corpus de domaines différents afin de valider les conclusions établies dans cet article. Par ailleurs, nous proposerons d'évaluer la qualité de l'enrichissement effectué avec *ExpLSA* en effectuant une validation par des mesures statistiques appliquées aux données du web. Nous envisageons également de mettre en œuvre d'autres méthodes afin d'ajouter des connaissances syntaxiques à LSA. Nous proposerons enfin d'utiliser des vecteurs sémantiques avec SYGMART en considérant un document comme produit d'un ensemble de concepts issus du thésaurus Larousse.

Références

Béchet N., Roche M. and Chauché J. (2007). Improving LSA by expanding the contexts. In proceedings of Context-Based Information Retrieval (CIR) workshop – CONTEXT'O7 (short paper), Roskilde University, Denmark, August 2007, pages 105-108.

Béchet N., Roche M. and Chauché J. (2008). *ExpLSA* : utilisation d'informations syntaxico-sémantiques associées à LSA pour améliorer les méthodes de classification conceptuelle. In proceedings of EGC'08.

Bourigault D. (2002). UPERY : un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus. In Actes de *TALN*, Nancy, pages 75–84,.

Brill E. (1994). Some advances in transformation-based part of speech tagging. In *AAAI, Vol. 1*, pages 722–727.

Chauché J. (1984). Un outil multidimensionnel de l'analyse du discours. In Proceedings of *Coling, Standford University*, California, pages 11–15, 1984.

Cornuéjols A. and Miclet L. (2002). Apprentissage artificiel, Concepts et algorithmes. Eyrolles.

Faure D. (2000). Conception de méthode d'apprentissage symbolique et automatique pour l'acquisition de cadres de sous-catégorisation de verbes et de connaissances sémantiques à partir de textes : le système ASIUM. *PhD thesis*, Université Paris-Sud, 20 Décembre.

Fitzgerald, W.J. (1999). The restoration of missing data using Bayesian numerical methods. *Proc ICASSP'99*, Vol. I-VI, 1999.

Kanejiya D., Kumar A. and Prasad S. (2003). Automatic evaluation of students' answers using syntactically enhanced LSA. In Proceedings of the *Human Language Technology Conference (HLT-AACL 2003) Workshop on Building Educational Applications using NLP*.

Landauer T. and Dumais S. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, pages 211–240.

Landauer T., Laham D., Rehder B. and Schreiner M. E. (1997). How well can passage meaning be derived without using word order? a comparison of latent semantic analysis and humans. In Proceedings of the *19th annual meeting of the Cognitive Science Society*, pages 412–417.

Moldovan D.I., Mihalcea R. (2000). Improving the search on the Internet by using WordNet and lexical operators. IEEE Internet Computing 4(1) 34 – 43.

Platt J. (1999). Fast training of support vector machines using sequential minimal optimization. In B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, Advances in Kernel Methods – Support Vector Learning, pages 185-208. MIT Press, Cambridge, MA.

Rehder B., Schreiner M., Wolfe M., Laham D., Landauer T. and Kintsch W. (1998). Using latent semantic analysis to assess knowledge: Some technical considerations. In *Discourse Processes*, volume 25, pages 337–354, 1998.

Roche M. and Chauché J. (2006). LSA : les limites d'une approche statistique. In Proceedings of *atelier FDC'06 (Fouille de Données Complexes), conférence EGC'2006*, pages 95–106,.

Turney P.D. (2001). Mining theWeb for synonyms: PMI– R versus LSA on TOEFL. In Proceedings of *ECML'01, Lecture Notes in Computer Science*, pages 491–502.

Voorhees E.M. (1994). Query Expansion using Lexical-Semantic Relations, in Proceedings of *ACM SIGIR'94*, Dublin.

Wiemer-Hastings P. and Zipitria I. (2001). Rules for syntax, vectors for semantics. In Proceedings of the *Twenty-third Annual Conference of the Cognitive Science Society*.

Annexe

SVM – LSA

Classe	Précision	Rappel	F-Score
france	0,56	0,67	**60,70%**
economie	0,8	0,83	**81,40%**
insolite	0,38	0,27	**31,60%**
sante	0,69	0,41	**51,30%**
monde	0,8	0,82	**80,50%**
politique	0,67	0,75	**70,50%**
culture	0,63	0,64	63,70%
sciences	0,57	0,48	**52,10%**
people	0,81	0,63	**70,50%**
technologies	0,83	0,78	**80,50%**
sport	0,9	0,91	**90,70%**

classes trouvées :	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*			***classes réelles***
	288	25	4	5	8	66	19	10	0	2	4	*a*	=	france
	34	**356**	1	1	6	10	1	6	0	12	3	*b*	=	economie
	15	3	**30**	1	18	1	18	11	4	5	5	*c*	=	insolite
	33	5	2	**50**	9	6	3	12	1	2	0	*d*	=	sante
	26	11	11	3	**404**	8	5	20	3	1	4	*e*	=	monde
	53	7	1	1	12	**228**	1	1	1	0	0	*f*	=	politique
	16	2	12	1	6	3	**114**	9	9	3	3	*g*	=	culture
	25	4	4	9	36	17	2	**94**	0	4	1	*h*	=	sciences
	9	0	8	0	5	2	16	0	**80**	8	0	*i*	=	people
	7	32	5	0	1	0	1	2	1	**177**	0	*j*	=	technologies
	12	0	1	1	3	1	0	0	0	0	**186**	*k*	=	sport

SVM – ExpLSA 0,9

Classe	Précision	Rappel	F-Score
france	0,55	0,63	58,60%
economie	0,78	0,83	80,50%
insolite	0,38	0,23	29,10%
sante	0,68	0,37	48,20%
monde	0,79	0,82	80,40%
politique	0,65	0,75	69,90%
culture	0,63	0,65	**63,90%**
sciences	0,53	0,44	48,30%
people	0,8	0,62	69,60%
technologies	0,8	0,76	77,70%
sport	0,9	0,92	91,30%

classes trouvées :	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*			***classes réelles***
	272	29	5	5	7	70	17	18	0	3	5	*a*	=	france
	33	**357**	0	1	8	11	1	6	0	10	3	*b*	=	economie
	15	3	**26**	3	21	1	17	9	5	7	4	*c*	=	insolite
	34	7	1	**46**	10	4	5	12	0	4	0	*d*	=	sante
	26	11	8	3	**407**	11	2	18	5	1	4	*e*	=	monde
	51	8	1	0	11	**230**	1	2	1	0	0	*f*	=	politique
	14	1	14	1	5	4	**115**	8	8	5	3	*g*	=	culture
	26	4	3	9	39	18	4	**87**	0	5	1	*h*	=	sciences
	11	0	5	0	4	2	19	0	**79**	8	0	*i*	=	people
	6	37	4	0	1	1	1	4	1	**171**	0	*j*	=	technologies
	10	0	1	0	4	1	0	0	0	0	**188**	*k*	=	sport

SVM – ExpLSA 0,6

Classe	Précision	Rappel	F-Score
france	0,49	0,62	54,30%
economie	0,71	0,81	75,50%
insolite	0,26	0,12	16,10%
sante	0,54	0,3	38,70%
monde	0,69	0,74	71,50%
politique	0,59	0,69	63,90%
culture	0,62	0,58	60,10%
sciences	0,39	0,3	33,90%
people	0,7	0,45	55,00%
technologies	0,75	0,63	68,40%
sport	0,91	0,85	87,80%

classes trouvées :	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*			***classes réelles***
	266	37	2	7	18	69	15	8	5	1	3	*a*	=	france
	31	**349**	1	0	11	18	1	6	0	11	2	*b*	=	economie
	28	4	**13**	4	22	2	10	15	2	9	2	*c*	=	insolite
	33	5	4	**37**	13	4	2	20	1	4	0	*d*	=	sante
	45	18	6	2	**368**	17	2	25	8	1	4	*e*	=	monde
	43	16	2	1	27	**211**	0	5	0	0	0	*f*	=	politique
	25	6	11	0	8	5	**103**	7	4	5	4	*g*	=	culture
	29	8	4	14	42	22	8	**59**	3	6	1	*h*	=	sciences
	23	0	4	1	13	2	16	2	**58**	8	1	*i*	=	people
	13	50	2	2	4	1	7	3	2	**142**	0	*j*	=	technologies
	12	1	1	0	8	4	1	2	0	2	**173**	*k*	=	sport

Table 3 : Fscore par classe et matrice de confusion pour l'algorithme SVM