~~L~~*SVM*A : au plus deux composants pour apparier des résumés à des articles

Yves Bestgen

UCL/CECL/IPSY, B-1348 Louvain-la-Neuve, Belgique
yves.bestgen@psp.ucl.ac.be

Résumé

Dans le cadre de la tâche 2 de DEFT2011 qui consiste en l'appariement d'articles scientifiques avec le résumé correspondant, une approche, dénommée LSVMA, est proposée. Elle est basée sur trois composants : l'analyse sémantique latente (LSA), les machines à support vectoriel (SVM) et l'assignation finale selon l'algorithme du meilleur d'abord (MA). Cette approche a permis d'apparier parfaitement les résumés aux articles. Des analyses complémentaires montrent que le composant LSA n'est pas indispensable pour relever efficacement le défi. Par contre, une optimisation de l'assignation effectuée par la SVM est nécessaire, à tout le moins pour les options et paramètres testés. Le caractère superflu de LSA pour la tâche proposée contraste avec le rôle qu'il joue dans les systèmes d'évaluation automatique de résumés. Cette recherche ne permet toutefois pas de décider si cette conclusion est spécifique au présent défi ou si elle peut être généralisée à d'autres tâches mettant en jeu l'évaluation automatique de résumés.

Abstract

Within the framework of DEFT2011 second task, which consists in the pairing of a scientific article with its corresponding abstract, an approach, called LSVMA, is proposed. It is based on three components: Latent Semantic Analysis, Support Vector Machines (SVM) and a final assignment step based on the best-first algorithm (*Meilleur d'Abord* : MA). This approach made it possible to pair the summaries with the articles without any error. Complementary analyses show that LSA is not necessary to take up the challenge effectively while an optimization of the assignment carried out by the SVM is necessary, at least for the options and parameters tested. The unnecessary character of LSA for this task contrasts with the role it plays in automatic summary grading. This study does not make it possible to decide whether this conclusion is specific to the present challenge or can be generalized to other tasks.

Mots-clés : Evaluation automatique de résumés, Machines à support vectoriel, Analyse sémantique latente, Problème d'affectation, Meilleur d'abord
Keywords: Automatic summary grading, Support Vector Machines, Latent Semantic Analysis, Assignment problem, Best-first

1 Introduction

La tâche 2 de DEFT2011 consiste en l'appariement d'articles scientifiques avec le résumé correspondant. Selon le point de vue adopté, de nombreuses approches sont envisageables pour relever ce défi. Si, par exemple, on s'appuie sur le fait que l'auteur d'un résumé est (très probablement) aussi l'auteur de l'article en question, des procédures développées afin d'identifier l'auteur d'un texte sont tentantes. Partir du postulat que nombre de résumés sont rédigés lorsque l'article est terminé et qu'ils réutilisent des passages de cet article conduit à se tourner vers des procédures efficaces dans le cadre de la détection de la "réutilisation" de passages dans différents textes. L'évidente parenté thématique entre le résumé et l'article conduit, quant à elle, à privilégier le recouvrement en termes de contenu. C'est, par exemple, le point de vue choisi par Foltz, Britt et Perfetti (Foltz, 1996) dans leur étude visant à déterminer au moyen d'une procédure automatique, basée sur l'analyse sémantique latente (LSA : Latent Semantic Analysis), quels textes ont le plus influencé les résumés produits après la lecture d'un grand nombre de textes abordant un même sujet.

L'approche évaluée dans ce rapport s'inscrit prioritairement dans cette troisième option parce qu'elle semble être la plus pertinente dans le cadre des travaux sur l'évaluation automatique de la qualité d'un résumé. Comme le soulignent les organisateurs de DEFT[1], évaluer un résumé est une question complexe et importante (Das, Martin, 2007). Elle est également au centre d'un champ de recherches très dynamique en éducation comme l'atteste le développement d'outils pour l'évaluation des résumés produits par des étudiants et celui de tutoriels visant à les aider à améliorer leurs résumés (Franzke et al., 2005 ; He et al., 2009 ; Kintsch et al., 2000 ; Miller, 2003 ; Olmos et al., 2009 ; Wade-Stein, Kintsch, 2004). Cet intérêt pour les résumés trouve son origine dans les processus cognitifs sous-jacents à cette activité qui conduisent l'étudiant à porter une attention toute particulière aux informations les plus importantes d'un texte et à intégrer celles-ci avec ses connaissances antérieures, deux processus primordiaux pour un apprentissage efficace par la lecture (Wade-Stein, Kintsch, 2004). Dans le cadre de ce défi, l'idée à l'origine de l'approche proposée est qu'une procédure qui fonctionne pour évaluer des résumés devrait donner une meilleure "note" au résumé du texte qu'aux résumés d'autres textes. La section suivante décrit cette approche. Les analyses et résultats obtenus sont présentés dans la troisième section. Ensuite, l'utilité des composants LSA et Assignation au meilleur d'abord pour l'efficacité de la procédure est évaluée au travers d'analyses complémentaires.

2 Description de l'approche

Pour évaluer la qualité de résumés produits par des étudiants, l'approche classique consiste à comparer ces résumés à un document de référence au moyen d'une mesure de similarité, les meilleurs résumés étant ceux qui ressemblent le plus au document de référence. Dans ces travaux, la similarité est habituellement obtenue par l'entremise d'une analyse sémantique latente, une technique mathématique qui vise à extraire un espace sémantique à partir de l'analyse statistique des cooccurrences dans un corpus de textes (Deerwester et al., 1990 ; Landauer et al. 1998)[2]. Le corpus employé pour extraire les dimensions sémantiques peut être composé d'un très grand nombre de textes censés être représentatifs des documents lus par des étudiants (Tasa corpus, 11 millions de mots) ou d'un corpus plus petit, mais composé de textes thématiquement similaires à ceux qui doivent être évalués (Kintsh et al., 2000). Il peut également être composé des seuls documents analysés (He et al., 2009). Quant au document de référence, il s'agit soit d'un résumé-idéal, le plus souvent rédigé par un expert du domaine sur lequel le texte porte, ou du texte qui devait être résumé. Lorsque le document de référence est le texte initial, la comparaison peut se faire sur la base du texte considéré comme un seul document ou sur la base des différentes sections ou des paragraphes qui le composent. Ces différentes options ont été combinées de très nombreuses manières (León et al., 2005), le cas le plus extrême étant probablement la technique proposée par He et al. (2009) qui découpe le résumé-cible et le résumé-idéal en phrases, procède à une analyse sémantique latente indépendante de ces deux micro-documents et obtient, sur cette base, les similarités entre les phrases qui composent les deux

[1] http://deft2011.limsi.fr/index.php?id=4&lang=fr

[2] LSA est aussi une des techniques employées pour générer automatiquement le résumé d'un texte, voir par exemple Gong et Liu (2001) ou Steinberge et Jezek (2004), mais ce sujet ne sera pas abordé ici parce que l'approche qui consiste à générer le résumé des articles et à ensuite le comparer aux résumés potentiels n'a pas été suivie.

documents. Les études qui ont comparé ces options mettent en évidence peu de différences entre elles (León et al., 2005 ; Olmos et al., 2009).

Ces observations laissent penser qu'une approche basée sur la comparaison des résumés aux différentes sections des textes en employant un espace sémantique spécifique à ce matériel devrait être efficace pour effectuer la tâche d'appariement. Cette tâche présente toutefois une spécificité par rapport à l'évaluation de résumés qui justifie une adaptation de la procédure : l'existence d'une correspondance biunivoque entre l'ensemble des résumés et l'ensemble des textes. Afin de tirer profit de cette particularité, l'approche classique qui s'appuie sur une mesure de similarité entre les résumés et les segments de textes a été remplacée par une procédure de classification supervisée qui apprend, à partir des informations issues de LSA, à classer correctement des paragraphes en fonction du texte dont ils ont été extraits et qui est ensuite appliquée à la catégorisation des résumés selon les mêmes principes (pour d'autres études qui combinent LSA et SVM, voir, par exemple, Bechet et al. (2008) ou Kwok (1998)). Il est à noter que la procédure supervisée employée ne requiert pas un échantillon d'apprentissage pour lequel l'appariement entre les textes et les résumés est connu. En effet, la procédure est appliquée aux paragraphes des textes pour lesquels l'appariement est toujours connu : chaque texte est considéré comme une catégorie et chaque paragraphe comme une instance de cette catégorie. La prédiction est faite sur les résumés qui sont catégorisés dans une des catégories correspondant à un texte.

2.1 Principaux composants

L'approche proposée, dénommée **L*SV*<u>*MA*</u>**, repose sur trois composants : l'analyse sémantique latente (Latent Semantic Analysis : LSA), les machines à support vectoriel (SVM) et l'affectation par la technique du Meilleur d'Abord (MA) qui tire profit de la relation biunivoque entre les résumés et les articles. Ces trois composants sont décrits dans la présente section.

2.1.1 LSA

Comme indiqué ci-dessus, LSA est une technique mathématique qui vise à extraire un espace sémantique à partir de l'analyse statistique des cooccurrences dans un corpus de textes. Le point de départ de l'analyse est une matrice qui contient le nombre d'occurrences de chaque terme dans chaque document, un document pouvant être un texte, un paragraphe, une phrase ou même une suite de mots d'une longueur arbitraire. Après normalisation (optionnelle) de la matrice de fréquences, celle-ci fait l'objet d'une décomposition en valeurs singulières. La matrice initiale (X) est décomposée en un produit de trois matrices (TSD') comme illustré à la figure 1. S est une matrice diagonale correspondant aux valeurs singulières et T et D sont des matrices orthogonales correspondant respectivement aux vecteurs singuliers pour les termes et pour les documents. En ne retenant que les *k* plus grandes valeurs singulières, on peut reconstruire une version compressée de la matrice originale dans laquelle seules les dimensions les plus importantes ont été conservées. Dans la suite des analyses, ce sont les *k* vecteurs singuliers de la matrice D qui sont employés comme traits dans la SVM afin d'apprendre à classer les paragraphes en fonction de l'article d'origine.

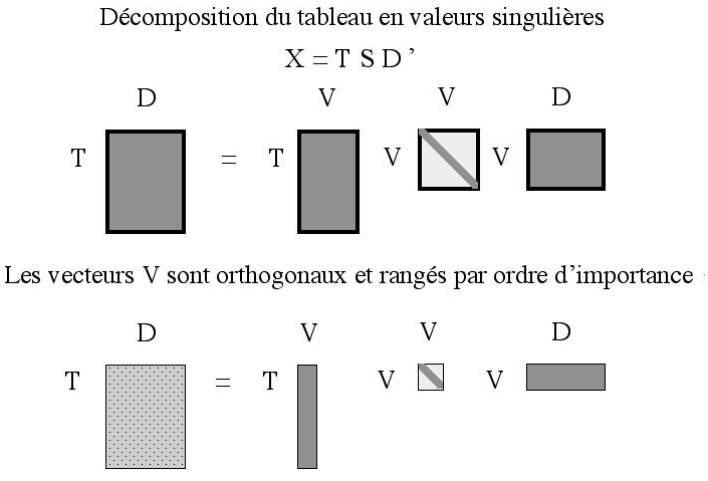

Figure 1 : Décomposition en valeurs singulières d'une matrice de fréquences Termes*Documents

Pour effectuer la décomposition en valeurs singulières de la matrice de fréquences (après pondération par la formule classique de log-entropie, Dumais, 1991), j'ai employé le programme *SVDPACKC*[3] (Berry et al., 1993), procédure LAS2 (*Single Vector Lanczos Method*).

2.1.2 SVM

L'algorithme d'apprentissage utilisé est une machine à support vectoriel (SVM) connue pour son efficacité en catégorisation supervisée de textes (Joachims, 2002). Dans sa forme la plus classique, cet algorithme apprend à classer un ensemble d'exemplaires en deux catégories sur la base de traits qui correspondent ici aux vecteurs singuliers issus de LSA. Dans le cas présent, le problème est multicatégoriel puisqu'il s'agit de catégoriser chaque paragraphe (lors de l'apprentissage) et chaque résumé (lors du test) dans une des catégories correspondant à chaque fois à un article. Plusieurs approches ont été proposées pour appliquer les SVM à ce type de situations comme l'emploi d'une série de classifieurs binaires (en "un contre tous" ou en "un contre un") ou d'une approche intrinsèquement multiclasse (Crammer, Singer, 2001). C'est cette dernière option qui a été choisie ici en raison de la simplicité de sa mise en pratique. Pour effectuer cette analyse, le logiciel SVM^{multiclass}[4] de Joachims (Joachims et al., 2009) a été employé. Le paramètre de régularisation *C*, qui détermine le rapport entre la taille de la marge et le nombre d'erreurs tolérées lors de l'apprentissage, a été fixé à 1 (mais voir la section 4 pour une analyse d'autres valeurs).

Une limitation des SVM pour le problème à résoudre est que la classification finale est effectuée de manière indépendante pour chaque résumé, ce qui ne garantit pas que chaque texte ne se verra affecter qu'un seul résumé. Le troisième composant tente d'optimiser l'approche en tirant profit de cette propriété de la tâche.

2.1.3 Appariement par le meilleur d'abord

Pour chaque résumé, la procédure SVM attribue, au travers des valeurs de décision, un score de comptabilité avec chacune des catégories et donc avec chaque article. On obtient donc une sorte de matrice de similarité Résumés*Textes et il s'agit d'apparier le plus correctement possible chaque résumé à un et un seul texte. Cette situation peut être vue comme un classique problème d'affectation dans lequel des tâches doivent être assignées à des agents pour un coût minimal (ou un bénéfice maximal). Le même genre de problème se rencontre en méthodologie de la recherche dans les études observationnelles qui visent à comparer deux groupes comme des personnes "malades" et des personnes "saines" (Rubin, 1973). Afin d'accroître l'efficacité des analyses, on y procède à l'appariement des participants du groupe cible à ceux du groupe contrôle afin de constituer les paires dont les membres sont les plus similaires possible. Différentes techniques ont été proposées par Rosenbaum et Rubin (1985). La plus simple consiste à choisir aléatoirement un membre du groupe cible et à l'apparier au membre du groupe témoin qui lui est le plus similaire. Ensuite, chaque membre de cette paire est retirée du groupe respectif et la procédure est réappliquée jusqu'à ce que tous les membres du groupe cible soient associés à un membre du groupe contrôle. Comme le souligne Rosenbaum (2010), une telle approche n'est pas nécessairement optimale au sens qu'elle ne garantit pas une similarité maximale globale entre les couples ainsi formés parce qu'elle ne prend pas en compte le fait que l'appariement effectué à une étape modifie les appariements possibles lors des étapes ultérieures. Dans certaines situations, elle est néanmoins aussi efficace que l'optimisation globale. On peut penser que le cas considéré ici fait partie de ces situations puisque l'objectif n'est pas de construire un appariement à partir de données disparates, mais bien de retrouver l'appariement pré-existant aux analyses. Pour cette raison, la procédure d'appariement par paire selon l'approche du plus proche disponible (*nearest available pair-matching* ou *greedy*, Hansen, 2004) en suivant l'ordre du *Meilleur d'Abord* (MA) a été utilisée. Elle consiste simplement à associer en premier lieu le résumé et l'article qui ont le meilleur score de compatibilité dans toute la matrice, à retirer ce couple de la matrice et à réitérer cette procédure jusqu'à ce que tous les couples aient été formés. Comme indiqué ci-dessus, cette procédure ne conduit pas nécessairement à une solution globalement optimale et est donc potentiellement optimisable au moyen, par exemple, de la méthode hongroise proposée par Kuhn pour résoudre les problèmes d'assignation (Rosenbaum, 2010).

[3] http://www.netlib.org/svdpack/svdpackc.tgz

[4] http://svmlight.joachims.org/svm_multiclass.html

2.2 Implémentation

Cette section décrit différents aspects de l'implémentation qui n'ont pas été discutés dans la section précédente. Il est à noter que les analyses décrites ci-dessous ont été réalisées séparément pour chaque revue puisqu'une balise spécifiant la revue dont a été extrait le document était associée à chaque résumé et à chaque article et que les règles du défi indiquaient que cette information pouvait être employée.

2.2.1 Prétraitement

Dans un premier temps, les documents ont été lemmatisés au moyen du programme *TreeTagger* de Schmid (1994). La suite des analyses a été effectuée sur les formes lemmatisées, sauf lorsque le mot était inconnu du tagger et, dans ce cas, la forme originale a été conservée. Les signes de ponctuation et les nombres, identifiés par le tagger, ont été supprimés.

Ensuite, les articles ont été segmentés en paragraphes. Cette segmentation est basée sur les balises *<titre>* et *<p>*. Les titres étant très courts, chacun d'entre eux a été systématiquement associé au paragraphe qui le suit directement. Pour les paragraphes eux-mêmes, la situation est un peu plus complexe parce que la balise *<p>* est présente, dans certains textes, au début de chaque ligne et qu'elle est également employée pour isoler des exemples. On a donc décidé de fixer une taille minimale de 75 mots aux paragraphes et de regrouper tout paragraphe trop petit avec celui ou ceux qui le suivent (à l'exception du ou des derniers paragraphes qui, s'ils n'atteignaient pas la longueur minimale, étaient réunis aux paragraphes qui les précèdent). Dans la suite, les unités ainsi formées sont appelées paragraphes. Aucune segmentation n'a été appliquée aux résumés.

Après la segmentation, tous les déterminants définis et indéfinis ont été supprimés. Par contre, tous les autres mots fonctionnels comme les pronoms ou les conjonctions, ont été conservés. Le résultat de cette étape est, pour chaque revue, une matrice de fréquence des termes dans tous les paragraphes et dans tous les résumés. C'est cette matrice, après application de la classique pondération log-entropie, qui a été soumise à LSA. Une autre option, légèrement plus complexe, aurait été de ne soumettre à la SVD que les paragraphes et de positionner ultérieurement les résumés dans cet espace (en calculant la somme pondérée des vecteurs représentant les termes qui les composent).

2.2.2 Options d'optimisation

L'implémentation présentée ci-dessus inclut une série de paramètres et d'éléments optionnels qui peuvent être vus comme autant d'occasions pour tenter d'optimiser la procédure (voir par exemple Bestgen (2004) pour une analyse de l'impact de ces paramètres sur l'efficacité de LSA). Parmi ceux-ci, un paramètre mérite une attention particulière parce qu'il est connu pour affecter l'efficacité d'une procédure basée sur l'analyse sémantique latente et qu'il affecte directement la procédure de catégorisation par la SVM : le nombre de dimensions de l'espace sémantique pris en compte (*k* dans la section 2.1.1). Trois cents est généralement considéré comme un optimum, tout particulièrement lorsqu'on traite un très grand corpus (Landauer et al., 2004). Toutefois, lorsque les analyses sont effectuées sur des corpus spécifiques (textes d'un contenu similaire au texte cible), un nombre plus réduit de vecteurs est conservé (Olmos et al., 2009). Dans le cas présent, le matériel de développpement pour une des revues n'est composé que de 1297 documents (paragraphes et résumés, voir Tableau 1), ce qui correspond à un matériel d'à peine 650 documents pour la phase de test. Il a donc été décidé de comparer l'efficacité de la procédure pour 300 vecteurs singuliers, mais aussi pour 200 et 100.

3 Analyses et résultats

3.1 Analyse du matériel de développement

Le tableau 1 présente le matériel qui a servi pour le développement de la procédure et ce pour les deux pistes du défi, la piste 1 celle qui porte sur le texte complet de l'article et la piste 2 pour laquelle

l'introduction et la conclusion de l'article ont été supprimées du texte à apparier. L'abréviation employée pour faire référence aux différentes revues dans la suite est donnée en dessous du nom de celle-ci.

		Piste 1		Piste 2	
Revue	Couples	Paragraphes	Termes différents	Paragraphes	Termes différents
Anthropologie et Sociétés (ANH)	60	1982	8952	1558	8439
Meta (MET)	59	2569	8652	2330	8500
Revue des sciences de l'éducation (SCI)	60	2427	7072	2078	7003
Études internationales (INT)	60	2546	7984	2004	7848
Études littéraires (LIT)	60	1748	9319	1297	8285

Tableau 1 : Description du matériel de développement

3.1.1 Efficacité de SVM^{multiclass} lors du développement : catégorisation des paragraphes

Le Tableau 2 présente le pourcentage de paragraphes classés correctement par la procédure SVM selon le nombre de vecteurs singuliers employés (Nvec). Il s'agit des valeurs obtenues lors de l'apprentissage et donc sans application d'une procédure de validation croisée puisque la vraie évaluation de l'approche sera effectuée au travers de l'appariement des résumés aux textes. On observe que les vecteurs qui occupent les rangs allant de 101 à 300 apportent une contribution non négligeable à la catégorisation puisque pour une des cinq revue le pourcentage de bien classés n'est que de 81% pour 100 vecteurs alors qu'il est de à 94% pour 300.

Piste	Nvec	ANH	MET	SCI	INT	LIT
1	100	92.58	80.03	93.78	94.85	94.97
	200	96.97	89.96	96.54	96.90	98.40
	300	98.13	94.08	97.73	97.84	98.91
2	100	93.32	80.60	93.41	94.06	94.45
	200	97.56	90.73	96.44	96.56	98.30
	300	98.59	93.73	97.64	97.95	99.23

Tableau 2 : Pourcentage de classifications correctes lors de l'apprentissage

3.1.2 Efficacité de la procédure lors du développement : appariement résumé-texte

Le Tableau 3 présente le pourcentage d'appariements corrects pour les deux pistes selon le nombre de vecteurs singuliers (Nvec) mis à la disposition de la SVM. On observe que la revue *Meta* est la seule à

poser problème à LSVMA et ce seulement pour 100 ou 200 vecteurs singuliers. Cette observation peut être mise en relation avec la moindre performance du classifieur SVM pour cette revue lors de l'apprentissage et suggère qu'il pourrait être pertinent d'accroître le nombre de vecteurs singuliers pris en compte par la SVM. Toutefois, l'obtention d'une performance parfaite avec 300 vecteurs singuliers pour chacune des deux pistes du défi a rendu peu pertinente toute tentative d'optimisation des paramètres.

Piste	Nvec	ANH	MET	SCI	INT	LIT
1	100	100	89.83	100	100	100
	200	100	100	100	100	100
	300	100	100	100	100	100
2	100	100	89.83	100	100	100
	200	100	96.61	100	100	100
	300	100	100	100	100	100

Tableau 3 : Pourcentage d'appariements corrects

3.2 Analyse du corpus de test

Le matériel de test pour les deux pistes a été analysé, au moyen de la procédure décrite ci-dessus et, donc, en faisant varier le nombre de vecteurs singuliers mis à disposition de la SVM (100, 200 et 300). Ces trois analyses ayant produit les mêmes appariements, une seule soumission a été envoyée pour chaque piste. Les résultats, transmis par les organisateurs du défi, indiquent que les appariements proposés par LSVMA sont tous corrects, un niveau de performance également atteint par d'autres équipes, et ce malgré la présence d'une revue qui n'était pas incluse dans le matériel d'entraînement.

4 Evaluation de l'utilité de deux des trois composants

Des trois composants de l'approche proposée, seul SVM est indispensable parce c'est ce composant qui assigne, au moins provisoirement, les résumés aux articles. Il serait bien sûr possible de s'en passer, par exemple en calculant une mesure de proximité entre les paragraphes et les résumés, mais il s'agirait là d'une tout autre approche. Les deux autres composants sont facultatifs. Au lieu d'employer les vecteurs singuliers obtenus par LSA, SVM peut utiliser directement les termes comme traits et il n'est pas prouvé que l'assignation des résumés aux textes effectuée par SVM, n'est pas optimale, ce qui rendrait le passage par la procédure d'assignation au meilleur d'abord inutile. Afin de déterminer si ces deux composants sont utiles ou non pour effectuer la tâche, des analyses complémentaires ont été menées.

4.1 Nécessité de LSA?

Une des principales raisons de l'emploi de LSA pour estimer la similarité entre des documents est que cette technique permet de considérer comme similaires des documents même si ceux-ci n'ont pas de mots en commun (Miller, 2003). Cette propriété est très importante pour l'évaluation automatique de résumés ou le développement de tutoriels visant à aider des étudiants à écrire de meilleurs résumés puisqu'on s'attend à ce que les étudiants emploient leurs propres mots lors de la rédaction. Dans le cadre de ce défi, il est, par contre, peu probable qu'un résumé contienne un vocabulaire différent du texte correspondant. Il s'ensuit que LSA n'est peut-être pas nécessaire pour réaliser efficacement la tâche d'appariement.

Pour évaluer l'impact de LSA sur l'efficacité, ce composant a été retiré de la procédure. La SVM a donc été appliquée à la matrice Paragraphes*Termes (pondérée par log-entropie) issue directement des

prétraitements. Ce sont donc les termes qui servent de traits pour l'apprentissage et non les vecteurs singuliers issus de LSA.

Ces analyses ont d'abord été effectuées sur le matériel de développement en employant plusieurs valeurs pour le paramètre *C* de la SVM. Avec *C* fixé à 1 (comme dans les analyses précédentes), cette version de la procédure obtient un pourcentage d'appariements corrects de 97.32% pour les deux pistes du défi (articles entiers et articles tronqués). Avec un *C* fixé à 100, on obtient plus de 99% d'appariements corrects et 100% pour *C* = 1 000 ou *C* = 5 000 et ce, toujours, pour les deux pistes[5].

Des résultats identiques ont été obtenus avec le matériel de test pour *C* = 1 000 et *C* = 5 000. Il s'ensuit qu'avec une valeur *C* suffisamment grande, la procédure sans le composant LSA est aussi efficace qu'avec ce composant et que, donc, celui-ci n'est pas nécessaire pour la tâche en jeu. C'est la raison pour laquelle le L de ~~L~~*SVM*A a été biffé dans le titre de l'article.

4.2 Nécessité de l'assignation au meilleur d'abord?

Pour évaluer l'utilité du composant d'assignation finale au meilleur d'abord, on a déterminé l'efficacité des procédures LSA+SVM (avec 300 vecteurs singuliers) et SVM sans LSA (pour *C* = 1 000 et *C* = 5 000) avant que le composant Assignation ne soit appliqué. Tant sur le matériel de développement que sur le matériel de test, aucune des procédures n'a donné lieu à une performance parfaite. Il s'ensuit que ce composant améliore bien l'efficacité de la SVM.

4.3 Discussion

Les résultats présentés dans cette section ne peuvent être considérés comme "définitifs" parce, comme indiqué ci-dessus, l'approche proposée inclut un grand nombre d'options et de paramètres et qu'aucune étude systématique de leur impact n'a été réalisée. Il est donc possible que le composant "assignation au meilleur d'abord", déclaré "nécessaire", ne le soit plus lorsqu'un jeu plus optimal de paramètres est employé. Par contre, le fait qu'un composant (LSA dans le cas présent) se révèle non indispensable n'est pas remis en question par cette argumentation. On notera toutefois qu'il est possible que l'analyse sémantique latente se révèle aussi efficace que l'approche présentée ici si elle employée, non au travers d'une procédure SVM, mais par des approches plus classiques d'évaluation de résumés et optimisée dans ce cadre (Olmos et al., 2009).

5 Conclusion

Dans le cadre de la tâche 2 de DEFT2011 qui consiste en l'appariement d'articles scientifiques avec le résumé correspondant, nous avons proposé une approche basée sur trois composants : l'analyse sémantique latente, une machine à support vectoriel et l'assignation finale selon l'algorithme du meilleur d'abord. Cette approche a permis d'apparier parfaitement les résumés aux articles, et ce pour les deux pistes du défi, celle qui porte sur le texte complet de l'article et celle pour laquelle l'introduction et la conclusion de l'article ont été supprimées du texte à apparier. Des analyses complémentaires ont montré que le composant LSA n'est pas indispensable pour relever efficacement le défi. Par contre, une optimisation de l'assignation effectuée par la SVM est nécessaire, à tout le moins pour les options et paramètres testés. Le caractère superflu de LSA pour la tâche proposée contraste nettement avec le rôle qu'il joue classiquement dans les systèmes d'évaluation automatique de résumés. Cette recherche ne permet toutefois pas de décider si cette conclusion est spécifique au présent défi ou si elle peut, en partie au moins, être généralisée à d'autres tâches mettant en jeu l'évaluation automatique de résumés.

[5] Les analyses présentées à la section 3 ont été répétées en faisant également varier le paramètre *C*. Pour le matériel de test, les résultats sont identiques à ceux rapportés à cette section. Pour le matériel de développement, l'emploi d'une valeur *C* très élevée permet d'améliorer l'efficacité de l'approche pour 100 vecteurs singuliers.

Remerciements

Yves Bestgen est chercheur qualifié du Fonds de la recherche scientifique (FNRS) de la Communauté Wallonie-Bruxelles (Belgique).

Références

BECHET, N., ROCHE, M., CHAUCHE, J. (2008). ExpLSA et classification de textes, Actes des *9es Journées internationales d'Analyse statistique des Données Textuelles*, 167-177.

BERRY, M.. DO, T., O'BRIEN, G., KRISHNA, V., VARADHAN, S. (1993). SVDPACKC: Version 1.0 User's Guide, *Tech. Rep. CS-93-194*, University of Tennessee, Knoxville, TN.

BESTGEN, Y. (2004). Analyse sémantique latente et segmentation automatique des textes, Actes des *7es Journées internationales d'Analyse statistique des Données Textuelles*, 171-181.

CRAMMER, K., SINGER, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines, *Journal of Machine Learning Research* 2, 265-292.

DAS, D., MARTINS, A.F.T. (2007). A survey on automatic text summarization. Literature Survey for the Language and Statistics II course at CMU.

DEERWESTER, S., DUMAIS, S.T., FURNAS, G.W., LANDAUER, T.K., HARSHMAN, R. (1990). Indexing by Latent Semantic Analysis, *Journal of the American Society for Information Science* 41, 391-407.

DUMAIS, S. (1991). Improving the retrieval of information from external sources. *Behavior Research Methods* 23, 229-236.

FOLTZ, P.W. (1996). Latent Semantic Analysis for text-based research. *Behavior Research Methods* 28, 197-202.

FRANZKE, M., KINTSCH, E., CACCAMISE, D., JOHNSON, N., DOOLEY. S. (2005). Summary Street ® : Computer support for comprehension and writing. *Journal of Educational Computing Research 33* , 53-80.

GONG, Y., LIU, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. Proceedings of the *24th annual international ACM SIGIR conference on Research and development in information retrieval*, 19-25

HANSEN, B.B. (2004). Full matching in an observational study of coaching for the SAT. *Journal of the American Statistical Association* 99, 609-618.

HE, Y., HUI, S.H., QUAN, T.T. (2009). Automatic summary assessment for intelligent tutoring systems. *Computers & Education* 53, 890–899.

JOACHIMS T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*, Dordrecht, Kluwer.

JOACHIMS, T., FINLEY, T., YU, C. (2009). Cutting-Plane training of structural SVMs, *Machine Learning* 77, 27-59.

KINTSCH, E., STEINHART, D., STAHL, G., LSA RESEARCH GROUP, MATTHEWS, C., LAMB, R. (2000). Developing summarization skills through the use of LSA-based feedback. *Interactive Learning Environments 8*, 87-109.

KWOK, J.T. (1998). Automated text categorization using support vector machine. Proceedings of *ICONIP'98*, 347–351.

LANDAUER, T.K., FOLTZ, P.W., LAHAM, D. (1998), An introduction to latent semantic analysis. *Discourse Processes* 25, 259-284.

LANDAUER, T.K., LAHAM, D., DERR, M. (2004). From paragraph to graph: Latent Semantic Analysis for information visualization. Proceedings of the *National Academy of Science* 101, 5214-5219.

LEON, J., OLMOS, R., ESCUDERO, I., CAÑAS, J., SALMERON, L. (2005). Assessing short summaries with human judgments procedure and Latent Semantic Analysis in narrative and expository texts. *Behavior Research Methods* 38, 616-627.

MILLER, T. (2003). Essay assessment with Latent Semantic Analysis. *Journal of Educational Computing Research* 29, 495-512.

OLMOS, R., LEON, J.A., BOTANA, G.J., ESCUDERO, I. (2009) New algorithms assessing short summaries in expository texts using latent semantic analysis, *Behavior Research Methods* 41, 944-950

ROSENBAUM, P.R. (2010). *Design of Observational Studies*. Springer, New York.

ROSENBAUM, P.R., RUBIN, D. (1985), Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. *The American Statistician* 39, 33–38.

RUBIN, D. B. (1973), Matching to remove bias in observational studies, *Biometrics* 29, 159-183.

SCHMID, H., (1994). Probabilistic part-of-speech tagging using decision trees. Proceedings of the *International Conference on New Methods in Language Processing*, 44-49.

STEINBERGER, J., JEZEK, K. (2004). Text summarization and singular value decomposition. *Lecture Notes in Computer Science* 2457, 245–254.

WADE-STEIN, D., KINTSCH, E. (2004). Summary Street: Interactive computer support for writing. *Cognition and Instruction* 22, 333-362.