Les machines à vecteurs supports dans la catégorisation de textes arabes

Karim Djelailia
Université 08 Mai 1945
Guelma
Département d'Informatique
Karim_Djelai@Yahoo.fr

Abdessalem Kelaiaia
Université 08 Mai 1945
Guelma
Département d'Informatique
Sam_Kelaiaia@yahoo.fr

Hayat Farida Merouani
Université Badji Mokhtar
Annaba
Département d'Informatique
Hayet_merouani@Yahoo.com

Résumé : Dans cet article, nous présentons l'influence de la radicalisation et de la réduction de l'espace de représentation dans la qualité des résultats de classification de textes arabes. Le codage des termes extraits adopté étant le codage TF-IDF avec une radicalisation légère (light stemming). La réduction des termes, pour réduire l'espace de représentation, est basée sur le principe de corrélation entre les attributs extraits. Nous utilisons les machines à vecteurs supports (SVM) comme méthode de classification, eu égard à sa robustesse dans la catégorisation de textes; meilleure performance en terme de précision et meilleur temps d'apprentissage. Nous comparerons les résultats obtenus avec stemming et ceux utilisant directement les termes extraits à partir des textes du corpus. Nous testerons en parallèle l'effet de la réduction de l'espace de représentation dans la qualité des résultats de classification. Les résultats sont très encourageants et comparables à ceux obtenus pour l'anglais sur le corpus Reuters avec une f-mesure se situant entre 82 et 86% sur un corpus moyen de 402 textes arabes.

Mots-clés: SVM, catégorisation de textes, corpus, sélection d'attributs, stemming, TF-IDF.

1 INTRODUCTION

De nos jours, l'information textuelle est abondante sur le web. Elle représente une masse de près de 80% de l'ensemble de l'information qui y circule. Cette information serait sans intérêt si notre capacité à y accéder n'est pas conséquente.
Pour les langues telles que l'anglais ou les langues européennes, de grands pas sont déjà franchis dans ce domaine. Cependant, beaucoup reste à faire pour la langue arabe.
Nous allons nous occuper, dans cet article de l'un des aspects de la recherche d'information, en l'occurrence, la catégorisation de texte ou la recherche thématique. Nous adopterons une démarche par apprentissage supervisé utilisant les machines à vecteurs support pour leur efficacité dans le traitement des données de grandes dimensions. La base d'apprentissage est un corpus en langue arabe de documents étiquetés. Il sera question de distinguer entre les résultats obtenus avec une radicalisation de termes (stemming) de ceux obtenus avec un prétraitement rudimentaire consistant seulement à une simple tokénisation. En plus de l'évaluation d'une méthode de réduction de l'espace de représentation des termes par une méthode basée sur un seuil de corrélation.

2 CATEGORISATION AUTOMATIQUE DE DOCUMENTS

La catégorisation de textes consiste en la recherche d'une relation fonctionnelle entre un ensemble de textes et un ensemble de catégories [Sebastiani, 2002]. Ceci est basé essentiellement sur un algorithme d'apprentissage.
Le processus de catégorisation est un système qui reçoit en entrée un texte et en sortie, lui associe une ou plusieurs catégories. Ceci est effectué en respectant un ensemble d'étapes. Ces étapes concernent la représentation des textes, le choix de l'algorithme d'apprentissage, et l'évaluation des résultats en vue de prévoir le degré de généralisation du classifieur ainsi construit.
Dans ce processus, deux phases sont à distinguer à savoir, l'apprentissage et la classification.
Pour la recherche des termes (que ce soit ceux du corpus d'entraînement ou ceux des textes à classer), on peut, soit procéder à extraire exactement les mots contenus dans les textes, soit radicaliser les mots contenus dans les textes en vue de réduire l'ensemble d'indexation car dans toutes les langues, plusieurs mots peuvent avoir la même racine et donc presque le même sens.

Pour la représentation des textes, la méthode la plus communément utilisée est de transformer le texte en vecteur (représentation vectorielle). Chacune des dimensions de ce vecteur est un terme du texte. Une collection de texte peut être rassemblée en une matrice dont les lignes sont les documents de la collection et les colonnes sont les termes qui apparaissent au moins une fois dans les documents. On note par t_i le terme i de la collection et d_j le document j de la collection. La fréquence du terme i dans le document j est notée w_{ij}.
Cette fréquence peut prendre plusieurs formes, la plus utilisée étant la fréquence TF-IDF [Salton, 1988],[Joachims, 1999]) dont voici le formalisme :

$$\forall i \in [1..|V|], \quad d^{i}_{tf-idf} = tf_i^{\,d} \times \log(\frac{|D|}{df_i}) \qquad (2.1)$$

La première valeur est égale à la fréquence du mot i dans le document d (notée tf^d_i pour « term frequency » : généralement on utilise les fréquences normalisées) et la seconde valeur est égale à log(|D| /df_i) où |D| est le nombre de documents du corpus et df_i est le nombre de documents qui contiennent le mot i (df signifie document frequency). Et |V| est le cardinal de l'ensemble des termes contenus dans le corpus.
Cette représentation présente l'avantage de ne pas favoriser les termes les plus fréquents. Ces derniers correspondent généralement au mots outils utilisés dans les langues telles que les prépositions de conjonction, de coordination, les adverbes, …etc, . On les trouve généralement dans presque la totalité des documents. Leur effet est atténué par la deuxième partie de la formule TF-IDF. De même les mots les moins fréquents qui eux correspondent en général aux mots rares ou mal orthographiés, donc très peu pertinents pour discriminer les documents, ne sont pas pris en compte. Leur effet est atténué par la première partie de la représentation TF-IDF.
Pour avoir une idée de la dimension de l'espace de représentation, voici la représentation du corpus du journal El hayat
Nombre de documents : 42591
Nombre de termes distincts : 444761
D'où une matrice en TF-IDF de 42591 lignes et 444761 colonnes.
Cette propriétés de grande dimensionnalité de l'espace de représentation des informations textuelles rend caduque l'application de la plus part des algorithmes d'apprentissage. D'où la nécessité de procéder à sa réduction

3 LES MACHINES A VECTEURS SUPPORTS

La notion de machines à vecteurs supports a été introduite dans [Vapnik et Cortes, 1995]. L'idée de base soutenant cette notion est la minimisation du risque structurel. C'est-à-dire, que l'hypothèse expliquant un ensemble fini d'exemples peut être recherchée dans un sous ensemble de l'ensemble d'apprentissage.
Les SVM conviennent aux problèmes d'apprentissage à grandes dimensions. C'est le cas, précisément, des documents textuels. Leur utilisation est axée sur une catégorisation binaire (appartenance ou non appartenance à une classe donnée) d'où la notion d'exemples positifs et d'exemple négatifs.
Les individus (documents) positifs sont séparés des individus négatifs par un hyperplan séparateur qu'on notera H.
On note H^+ l'hyperplan parallèle à H et qui contient l'individu positif le plus proche de H et H^- l'individu négatif le plus proche de H comme illustré dans la figure 3.1

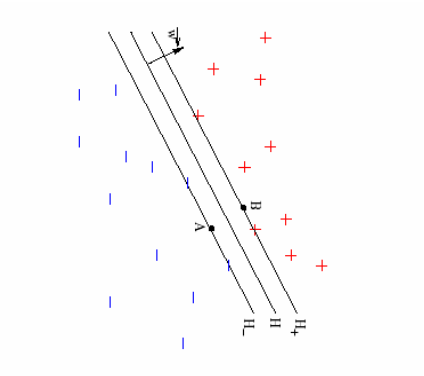


Fig 3.1. Illustration de la notion d'hyperplan séparateur

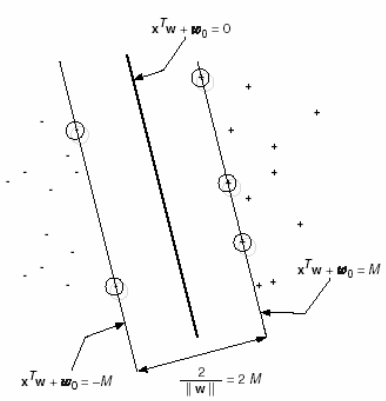


Fig 3.2. Illustration de la notion de marge

La théorie d'apprentissage statistique développée par Vapnik en 1998 [Vapnik et Cortes, 1995], démontre que nous pouvons définir un hyperplan (relatif à l'ensemble d'apprentissage) et possédant deux propriétés essentielles.

- Il est unique pour chacun des ensembles de données linéairement séparables
- Le risque de sur-apprentissage est le plus petit qui soit, relativement à n'importe quel autre hyperplan séparateur.

Nous définissions, la marge du classifieur comme la distance séparant l'hyperplan et les exemples les plus proches.
L'hyperplan optimal est celui qui possède la plus grande marge. Comme le démontre la figure 3.2

Le problème posé est un problème de résolution d'équation quadratique sous contraintes et revient à maximiser le dual du lagrangien dont l'équation est donnée par

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left\langle \vec{d_i}, \vec{d_j} \right\rangle \qquad (3.1)$$

Où les α_i sont les multiplicateurs de Lagrange
Les points ayant $\alpha_i > 0$ sont les vecteurs supports ce qui revient à chercher seulement ces points qui sont nécessaires à l'apprentissage.

4 EXPERIMENTATIONS

L'expérimentation que nous avons menée est basée sur un corpus diffusé sur le Web par Latifa Sulaiti [Al-Sulaiti et Atwell, 2004] en langue Arabe baptisés par ses auteurs (CCA : Corpus of contemporary arabic). Ce corpus est composé de 402 textes répartis en 14 catégories

4.1 Prétraitement

Nous avons procédé tout d'abord à la phase de prétraitement consistant en la tokénisation (casser les textes et extraire les mots) des textes du corpus après les avoir débarrassés des balises XML.
Nous avons opté pour la translittération de Tim Buckwalter.
Nous avons aussi procédé à l'élimination de tous les stops words à partir d'une liste de 131 mots
Le résultat de ces opérations est une base d'index bruts
Nous avons aussi préparé une base d'index radicalisés en utilisant un stemming (al-stem de Kareem darwish)

4.2 Expérimentations réalisées :

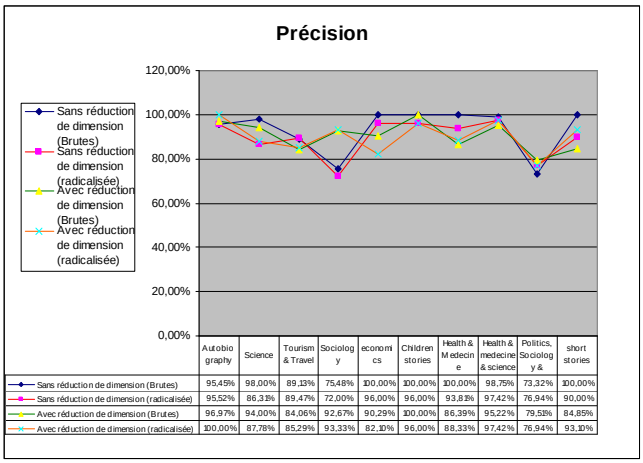
Pour l'expérimentation nous avons construit 10 classifieurs bi-classe sur les 8 classes les plus peuplées du corpus c'est-à-dire contenant plus de 25 documents en plus de deux classes que nous avons montées sur les

classes du corpus, en l'occurrence la classe « science, santé et médecine » et la classe « politique, sociologie et religion ».
Chaque classifieur a été bâti sous quatre aspects :

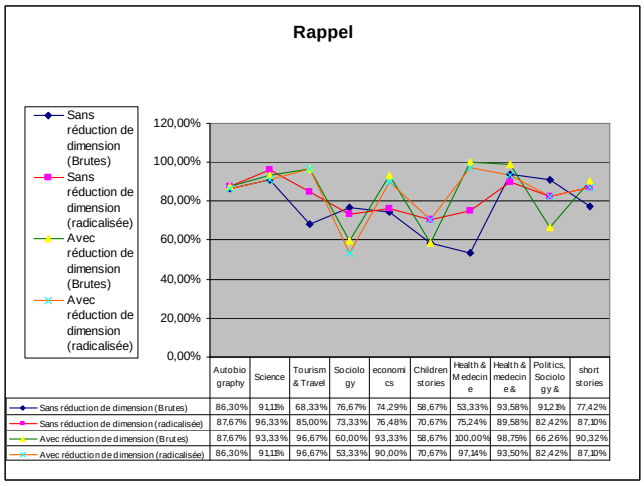
1- en données brutes,
2- en données radicalisées,
3- en données brutes avec réduction de dimension,
4- en données radicalisées avec réduction de dimension.

Nous avons étiqueté les documents appartenant à la classe des documents pertinents par « +1 » et nous avons pris autant de documents aléatoirement dans le restant du corpus pour construire la classe « -1 » de document non pertinents pour la classe considérée

5 RESULTATS ET DISCUSSIONS



	Autobiography	Science	Tourism & Travel	Sociology	economics	Children stories	Health & Medecine	Health & medecine & science	Politics, Sociology &	short stories
Sans réduction de dimension (Brutes)	95,45%	98,00%	89,13%	75,48%	100,00%	100,00%	100,00%	98,75%	73,32%	100,00%
Sans réduction de dimension (radicalisée)	95,52%	86,31%	89,47%	72,00%	96,00%	96,00%	93,81%	97,42%	76,94%	90,00%
Avec réduction de dimension (Brutes)	96,97%	94,00%	84,06%	92,67%	90,29%	100,00%	86,39%	95,22%	79,51%	84,85%
Avec réduction de dimension (radicalisée)	100,00%	87,78%	85,29%	93,33%	82,10%	96,00%	88,33%	97,42%	76,94%	93,10%



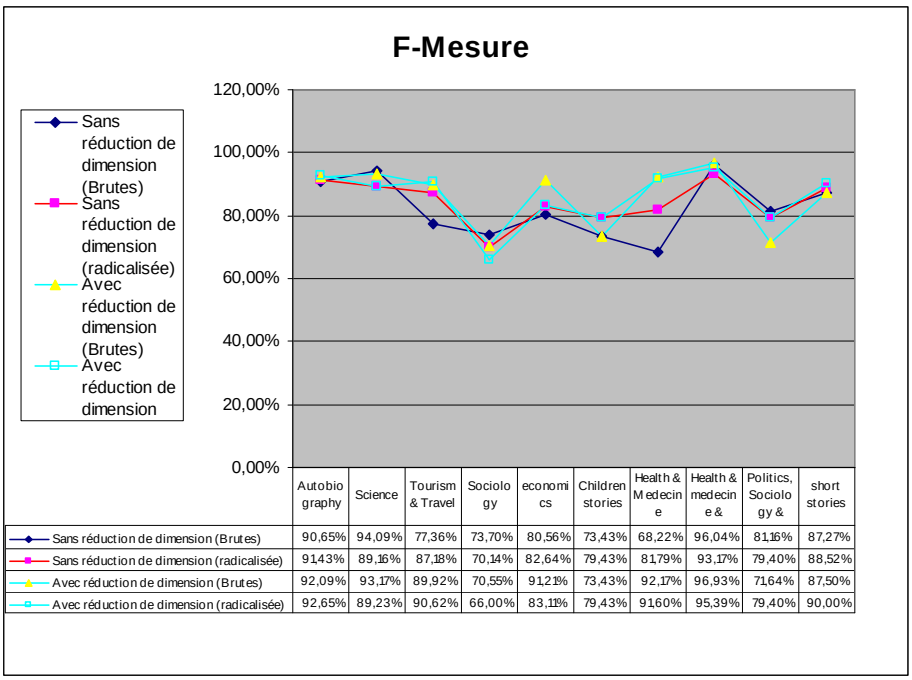
	Autobiography	Science	Tourism & Travel	Sociology	economics	Children stories	Health & Medecine	Health & medecine &	Politics, Sociology &	short stories
Sans réduction de dimension (Brutes)	86,30%	91,11%	68,33%	76,67%	74,29%	58,67%	53,33%	93,58%	91,21%	77,42%
Sans réduction de dimension (radicalisée)	87,67%	96,33%	85,00%	73,33%	76,48%	70,67%	75,24%	89,58%	82,42%	87,10%
Avec réduction de dimension (Brutes)	87,67%	93,33%	96,67%	60,00%	93,33%	58,67%	100,00%	98,75%	66,26%	90,32%
Avec réduction de dimension (radicalisée)	86,30%	91,11%	96,67%	53,33%	90,00%	70,67%	97,14%	93,50%	82,42%	87,10%

5.1 Précision

On remarque que la précision se dégrade avec le stemming
Pour les données sans réduction de dimension cette perte est de : 3,66 %
Pour les données avec réduction de dimension Il y a un petit peu de perte de précision de l'ordre de 0,37 % entre les données radicalisées et les données non radicalisées.
On observe tout de même une perte de précision globale de 0,97% entre les données sans réduction de dimension et les données avec réduction de dimension
Si le stemming nuit à la précision, la réduction de dimension en fait plus.

5.2 Rappel

On remarque que le rappel augmente avec le stemming ce qui confirme l'intuition. Il est amélioré de 5,29% sur les données sans réduction de dimension et de 0,32 % sur les données avec réduction de dimension
Aussi, il est à noter que le rappel augmente significativement avec la réduction de dimension. Cette augmentation est de 4,93 %
La réduction de dimension et le stemming ont un effet positif sur le rappel



	Autobiography	Science	Tourism & Travel	Sociology	economics	Children stories	Health & Medecine	Health & medecine &	Politics, Sociology &	short stories
Sans réduction de dimension (Brutes)	90,65%	94,09%	77,36%	73,70%	80,56%	73,43%	68,22%	96,04%	81,16%	87,27%
Sans réduction de dimension (radicalisée)	91,43%	89,16%	87,18%	70,14%	82,64%	79,43%	81,79%	93,17%	79,40%	88,52%
Avec réduction de dimension (Brutes)	92,09%	93,17%	89,92%	70,55%	91,21%	73,43%	92,17%	96,93%	71,64%	87,50%
Avec réduction de dimension (radicalisée)	92,65%	89,23%	90,62%	66,00%	83,11%	79,43%	91,60%	95,39%	79,40%	90,00%

5.3 F-mesure

On remarque que la f-mesure, compromis entre la précision et le rappel, est améliorée avec le stemming
Pour les données sans réduction de dimension ce gain est de : 2,04 %
Pour les données avec réduction de dimension ce gain est de :
* 3,61 % pour les données brutes
* 1,45 % pour les données radicalisées
On remarque tout de même qu'il n'y a aucun gain dans le cas de réduction de dimension entre les données brutes ou radicalisées
Conclusion : Si on ne procède pas à une réduction de dimension, Le stemming améliore la f-mesure

5.4 Autres observations

Le stemming ou radicalisation que nous avons opéré sur les textes bruts a permis la compression à plus de 60% sur les textes bruts sans réduction de dimension et à près de 45 % sur les textes après réduction de dimension. Avec le stemmer de porter appliqué au corpus Reuters-21578, le taux de compression est de 61% (15247 racines sur une base d'index de 39289) ce qui laisse à penser que al-stem est un outil aussi efficace pour la langue Arabe que l'est celui de Porter sur l'Anglais.
Pour la réduction de dimension, on note que près de 4 attributs sont corrélés sur les textes bruts et près de 3 attributs sur les textes radicalisés. La réduction de dimension permet d'obtenir des dimensions plus réduites à près de 3/4 sur les textes bruts et près de 2/3 sur les textes radicalisés. Sur de grandes bases d'apprentissage cette compression est très significative car elle contribue à une amélioration significative du temps de construction des classifieurs ainsi que les ressources nécessaires telles que la mémoire et les processeurs. Ce gain est très important pour des applications temps réel.

6 DISCUSSION GENERALE

On remarque tout de même que pour la langue Arabe sur le corpus CCA, les résultats coïncident avec ceux obtenus pour le corpus Reuters 21578.
Ceci nous mène à conclure que l'Arabe ne présente pas de particularité dans la phase d'apprentissage. Ses seules particularités résident dans la préparation des données (prétraitement).
A l'issue des expérimentations que nous avons menées sur le corpus CCA avec les machines à vecteurs supports et en utilisant la technique de stemming pour la radicalisation des termes, nous pouvons conclure ce qui suit :

Pour la langue Arabe, le stemming est une technique de représentation qui apporte une amélioration lors de la classification des documents comme dans les autres langues déjà testées. Cependant pour des applications favorisant la précision, cette technique contribue au bruitage du rendu par des documents non pertinents.
La réduction de dimension, si elle contribue à l'amélioration du temps de construction de classifieurs, elle possède les mêmes effets que le stemming au niveau de la précision. Quoiqu'en général, la f-mesure, se trouve améliorée dans les deux cas.

7 CONCLUSION ET PERSPECTIVES

Nous avons démarré de l'à priori que, vu les particularités de la langue Arabe, les méthodes de classification basées sur la représentation en sacs de mots seraient peut-être inefficaces et donneraient des résultats médiocres et on serait alors, amené à penser que seules les méthodes se basant sur une analyse morphosyntaxique sont prometteuses. Ce constat s'est avéré faux car à l'issue des résultats que nous avons obtenus nous pouvons conclure que les méthodes d'extraction d'attributs testées sur les autres langues sont adaptables à la langue Arabe et que seuls les prétraitements pour cette langue sont d'une complexité avérée. Les résultats des traitements, quant à eux, sont comparables à ceux obtenus sur les autres langues. Nous continuons à penser malgré tout que notre travail serait d'un intérêt avéré si nous le déroulerons sur un corpus plus important.

De ce fait, l'une de nos perspectives futures seraient d'élaborer un moyen de construction automatique de corpus en langue Arabe afin de le mettre à la disposition des équipes de recherche en « Recherche d'information ».

En ce qui concerne la classification automatique de textes Arabes, plusieurs issues sont encore ouvertes à savoir :

- **la représentation de documents** : Il n'a pas encore été prouvé que la représentation par les mots est la meilleure pour la langue Arabe. Il serait alors intéressant de tenter d'expérimenter d'autres représentations telles que les phrases ou les concepts guidés par une ontologie.

- **l'hybridation de méthodes** : Il serait peut être intéressant de penser à utiliser de façon complémentaire les informations tirées à partir d'une analyse morphosyntaxique des documents et ceux obtenu à travers les méthodes de représentation numérique en vue de construire des classfieurs et évaluer l'apport de cette hybridation.

BIBLIOGRAPHIE

[Al-Sulaiti, et Atwell, 2004] L. Al-Sulaiti et E. Atwell (2004).The Design of a Corpus of Contemporary Arabic. International Journal of Corpus Linguistics, vol. 11, forthcoming. 2006.
[Joachims, 1999] T. Joachims (1999). Transductive inference for text classification using support vector machines. In Bratko, I. et Dzeroski, S., editors, Proceedings of ICML-99, 16th International Conference on Machine Learning, PP. 200–209, Bled, SL. Morgan Kaufmann Publishers, San Francisco, US.
[Salton, 1988] G. Salton et C. Buckley (1988). Term-weighting Approaches in Automatic Text Retrieval. Information Processing and Management, 24(5), PP ; 513-523.
[Sebastiani, 2002] F. Sebastiani (2002). Machine learning in automated text categorization. ACM Computing Surveys, 34(1): PP. 1–47.
[Vapink et Cortes, 1995] V. Vapnik et C. Cortes (1995). Support vector networks. Machine Learning, 20: PP. 273–297.