L'organisation des connaissances et la recherche d'information textuelles par l'aplication des méthodes statistiques

Saeed RAHEEL

saeed.raheel@gmail.com

Doctoran en SIC - ELICO -ENSSIB, Université Lumière Lyon 2

***RÉSUMÉ**. Avec la globalisation du monde et l'évolution exponentielle du nombre d'internautes ainsi que les ressources en ligne s'introduit la nécessite indispensable de trouver des solutions adaptées pour l'organisation de cette immense et gigantesque ressource de connaissances appelée « internet ». C'est là où l'hybridation entre l'informatique, les sciences de l'information et de la communication, les statistiques ainsi que la linguistique s'avère être très utile. En effet, une approche pertinente très efficace issue de cette hybridation est la « classification automatique des documents ». Depuis quelques décennies, les chercheurs se sont intéressés à cette question. On retrouve des travaux portant sur ce sujet depuis le début des années 1960. Très peu des travaux se rapportent à la classification automatique de documents arabes et aucun d'entre eux n'utilise la technique de «Boosting ». Cet article vise à explorer l'utilisation de cette technique et son efficacité en vue de la classification automatique de documents arabes.*

***MOTS-CLÉS :** Fouille de textes, classification automatique, documents arabes, apprentissage artificiel, Boosting.*

***ABSTRACT:** With the world's globalization, the exponential evolution in the number of internet users as well as the online resources' mass emerges the inevitable need to come up with unorthodox and adapted methods to organize this gigantic resource of information named the "internet". This is where the hybridization between computer science, information and communication sciences, statistics, and linguistics proves to be very useful. In fact, a very efficient aproach of this hybridization is the «automatic classification of documents ». Since few decades, researchers have started working on this subject. We can easily find traces of their work since the beginning of the 1960s. We can find, however, very few resources dealing with documents written in Arabic characters and none of which uses the of «Boosting». technique This paper aims at exploring this technique and its efficiency concerning the automatic classification of Arabic document.*

***KEYWORDS :** Text mining, automatic classification, Arabic texts, machine learning, Boosting.*

1. Introduction

« L'organisation des connaissances est un domaine de recherche pluridisciplinaire et les Sciences de l'information n'en ont pas le monopole. D'autres disciplines y produisent concepts, formalismes et applications. Citons l'informatique, la philosophie, la linguistique, les sciences cognitives, les sciences de l'éducation et plus généralement les sciences humaines et sociales »[1]. Récemment, les chercheurs ont commencé à donner beaucoup d'importance aux traitements et à l'organisation des données textuelles multilingues, et cela pour plusieurs raisons : le nombre croissant de ces données mises en ligne, le développement de l'infrastructure de communication et de l'internet, la progression constante du nombre de personnes connectées au réseau mondial et dont la langue maternelle n'est pas l'anglais. Ces raisons ont incité les chercheurs à trouver des nouvelles méthodes de traitement automatique pour organiser l'immense volume de ces données. Pour cela, un nouveau paradigme basé sur l'apprentissage artificiel ou automatique (en anglais, *Machine Learning*) vient remplacer les anciennes approches. Selon Mitchell, «*L'aprentissage artificiel est un sous-domaine de l'intelligence artificielle qui s'intéresse à conférer aux machines la capacité de s'améliorer à l'accomplissement d'une tâche, en interagissant avec leur environnement*»[2].

L'apprentissage artificiel se divise principalement en deux façons : l'apprentissage supervisé et l'apprentissage non supervisée. La distinction principale entre ces deux approches, en général, est que dans l'approche supervisé le système dispose déjà des catégories dans lesquelles il doit classer ses documents tandis que dans l'approche non-supervisée le système doit analyser ses documents pour constituer des groupes de documents dont les profils sont les plus similaires entre eux et les plus dissimilaires avec les profils des documents des autres groupes. On appelle ces profils des amas (en anglais *clusters*). C'est dans l'approche dite supervisée que s'inscrit la façon dont nous abordons aujourd'hui le problème de la classification automatique de documents, et notamment les documents rédigés en arabes.

2. Définition formelle de la classification de texte

La classification de documents est définie comme la tâche de trouver une liaison entre un ensemble de documents D et un ensemble de catégories C. Formellement, la classification consiste à assigner une valeur v_{ij} de l'ensemble {0,1} à chaque entrée (d_i,c_j) de la matrice décisionnelle présentée dans le tableau 1 ci-dessous.

C \ D	d_1	…	d_m
c_1	v_{11}	…	v_{1m}
⋮	…	…	…
c_n	v_{n1}	…	v_{nm}

Tableau 1 : Matrice décisionnelle

Nous représentons ce processus, plus formellement, comme la fonction

[1] Yolla Polity, « L'organisation des connaissances en France : état des lieux », Communication aux Journées d'étude du Chapitre français de l'ISKO. Lille, 16-17 octobre 1997 ; Publiée dans les actes : *Organisation des connaissances en vue de leur intégration dans les systèmes de représentation et de recherche d'information*, éd. par J.Maniez et W. Mustafa Elhadi, Université de Lille, 1999, pp.n367-376

[2] Mitchell, T. M. (1997). *Machine Learning,* Computer Science. New York: McGraw-Hill., Preface, p. XV.

$\Phi : DxC \rightarrow \{0,1\}$ [1]

où,

C = {$c_1 \ldots c_n$} est l'ensemble prédéfini des catégories,

et D = {$d_1, \ldots, d_n$} est l'ensemble des documents à catégoriser.

Une valeur de 1 pour v_{ij} signifie que le document d_j doit être placé sous la catégorie c_j, bien qu'une valeur de 0 signifie le contraire. Le but de la classification de texte est de construire le modèle qui associe un document à une ou plusieurs catégories telle que la décision donnée par ce modèle satisfait le maximum possible la vraie fonction Φ. Nous appelons cette liaison fonctionnelle modèle de prédiction estimée par un apprentissage artificiel. Pour ce faire, l'algorithme doit se disposer d'un ensemble de textes préalablement étiquetés, que l'on appelle ensemble d'apprentissage, à partir duquel le modèle de prédiction le plus fidèle est estimé c'est-à-dire le modèle qui génère le moins d'erreur en prédiction.

3. La classification automatique des documents pour l'organisation des connaissances

La catégorisation de textes comporte un choix de techniques d'apprentissages (ou classifieur) disponibles. Parmi les méthodes d'apprentissage les plus souvent utilisées figure l'analyse factorielle discriminante (Bardos, 2001), la régression logistique (Hilbe, 2009), les réseaux de neurones (David, 2003), les plus proches voisins (Shakhnarovish, 2005), les arbres de décisions (Govindarajan, 2007), les réseaux bayesiens (Rich, 2007), les machines à vecteurs supports (Pilászy, 2005) et, plus récemment les méthodes de boosting (Schapire, 2002). La différence entre ces méthodes d'apprentissage est leur mode de construire leurs classifieurs ces classifieurs sont-ils construits manuellement ou bien automatiquement par induction à partir des données? En plus, est-ce que leur modèle apris est compréhensible, ou bien il s'agit d'une fonction numérique calculée à partir de données servant d'exemples ? Le choix du classifieur réside toujours dans l'objectif final à atteindre.

Selon Saracevic[3], "*les sciences de l'information sont un champ dévolu à des recherches scientifiques et une pratique professionnelle concernées par les problèmes d'une communication efficace du savoir et des documents qui contiennent ce savoir, cela dans un contexte social institutionnel et/ou individuel d'usages et de besoins d'information et avec une utilisation des technologies modernes de l'information*". Une grande partie des connaissances sont transmises, depuis longtemps, d'une génération à l'autre sous forme textuelle, comme par exemple, les ouvrages, revus, articles, etc. Pour que l'accès à ces documents soit le plus rapide possible, cela nous incite à les organiser. On pourrait, évidemment, demander à un humain de les classer manuellement. Cette tâche s'avère colossale s'il fait face à des centaines, voire des milliers de documents. Il serait alors très intéressant de recourir à un outil qui assignerait ces textes à un ensemble prédéfini de catégories d'une façon automatique. C'est clairement là le but de la classification automatique des textes.

On peut facilement imaginer l'utilité d'un tel processus et ses applications multiples. Par exemple,

[3] Saracevic, T. Information Science. Journal Of The American Society For Information Science. 50(12):1051–1063, 1999, p. 5.

une fois les documents catégorisés, un outil peut les acheminer facilement vers des destinataires intéressés par leurs sujets. Un autre exemple serait l'indexation automatique des documents, ouvrages, revues, etc. dans une bibliothèque où ces derniers sont assignés des labels, chacun selon son emplacement dans la bibliothèque. Considérons aussi le filtrage d'articles provenant d'une agence de presse ou à la détection de courriel indésirable. La classification automatique de textes pourrait aussi faciliter la recherche de pages Web par leur triage dans une hiérarchie de style Yahoo! En fait, la liste ne s'arrête pas ici mais par contre loin de là, ce qui indique sans aucun doute que la classification automatique contribue à l'évolution de la communication des connaissances laquelle contribue à son tour à l'intelligence collective.

4. La classification d'un document: comment ?

La classification de documents, en général, se divise en plusieurs étapes : la phase du traitement des textes, le choix de la façon par laquelle on va représenter ces documents, la réduction de la taille du vocabulaire, le choix de l'algorithme d'apprentissage, et enfin, l'évaluation des résultats obtenus.

4.1 La phase du traitement du texte

La première étape pour la classification d'un document commence par une phase de traitement. Ce traitement est composé de la manière suivante :

1. La conversion du document, que ce soit un HTML, PDF, XML, etc. en un document plein texte.
2. Tous les chiffres et les marques de ponctuation sont supprimés.
3. La dévoyellation complète ou partielle de tous les mots arabes voyellés
4. Tous les « mots vides » sont enlevés. Les « mots vides » sont les prépositions, les articles, les conjonctions, etc. qui ne portent pas de sens.

4.2 La représentation d'un document

A ce jour-là, aucun algorithme d'apprentissage n'est capable de traiter directement les textes. C'est pourquoi une étape dite de *représentation* est indispensable. Dans cet article nous avons choisi d'utiliser la représentation vectorielle, largement utilisée, dans laquelle chaque texte est représenté par un vecteur de n termes[4] calculés. Ainsi, un document d_i se transforme en un vecteur $v_j = \{p_{1i}, p_{2i}, \cdots, p_{|T|n}\}$ où, T est l'ensemble des descripteurs qui apparaissent au moins une fois dans l'espace d'apprentissage et p_{ki} correspond au poids du terme t_k *appartenant* au vocabulaire d'un document d_i.

4.3 La réduction de la taille du vocabulaire (RTV)

Si nous utilisons tous les mots présents dans les documents de l'espace d'apprentissage, nous nous

[4] Dans le contexte de cet article, les termes « terme », « mot », « attribue », ou « descripteur » sont interchangeables et signifient la même chose, à savoir un mot faisant parti du contenu d'un document.

retrouvons face un espace vectoriel ayant une dimension très large. Le traitement d'un tel espace vectoriel demanderait beaucoup de mémoire et de temps de calcul et pourrait nous empêcher d'utiliser des algorithmes de classification plus complexes. Le problème pourrait dans certains cas devenir non soluble. Utiliser tous ces mots influencerait aussi négativement la précision de la classification. Pour cela, nous avons appliqué trois techniques de réduction de la taille du vocabulaire basée sur la sélection des termes dont la première est la méthode de *Gain d'Information (GI)*. Cette méthode vise à mesurer le nombre de bits d'information obtenus pour la classification d'un document en sachant la présence ou l'absence d'un mot dans un document. Soit $\{c_1, c_2, ..., c_k\}$ l'ensemble des catégories. On calcule le gain d'information $GI(m)$ d'un mot m comme :

$$GI(m) = -\sum_{j=1}^{k} P(c_j) \log P(c_j) + P(m) \sum_{j=1}^{k} P(c_j|m) \log P(c_j|m) + P(\bar{m}) \sum_{j=1}^{k} P(c_j|\bar{m}) \log P(c_j|\bar{m}) \quad [2]$$

Ici,

$P(c_j)$ représente la proportion des documents appartenant à la classe c_j,
$P(m)$ représente la proportion des documents contenant le mot m,
$P(c_j|m)$ représente la proportion des documents appartenant à la classe c_j et contenant au moins une seule fois le mot m,

et, $P(c_j|\bar{m})$ représente la proportion des documents appartenant à la classe c_j et ne contenant pas le mot m.

Le gain d'information est calculé pour chaque mot appartenant à l'espace d'apprentissage et pour ceux qui ont un gain d'information inférieur à un certain seuil. Nous avons également utilisé durant nos expérimentations, les méthodes de Chi Quarré (χ^2) et le Rapport de Gain (*RG*). La statistique de χ^2 mesure le manque d'indépendance entre un mot m et une classe c_j. Sa formule est désignée par :

$$X^2(m, c_j) = \frac{N \times (AD - CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D)} \quad [3]$$

Où,

A correspond au nombre de documents appartenant à la class c_j et contenant le mot m,

B correspond au le nombre de documents contenant le mot m mais n'appartenant pas à la classe c_j,

C correspond au le nombre de documents appartenant à la classe c_j mais ne contenant pas le mot m,

D correspond au nombre de documents n'appartenant pas à la classe c_j et ne contenant pas le mot m,

et N est le nombre total de documents.

La méthode GI a le biais naturel de favoriser les attributs ayant plusieurs valeurs. Le rapport de gain, par contre, vise à les pénaliser en incorporant un terme, qu'on appelle *le partageur d'information* (le dénominateur de l'équation ci-dessous), qui est sensible a l'uniformité du partitionnement des

données selon chaque valeur *v* d'un certain attribut *A*.

$$RG(N,A) = \frac{GI(N,A)}{-\sum_{i=1}^{v} \frac{|N_v|}{|N|} \log_2 \frac{|N_v|}{|N|}} \quad [4]$$

N_v est le nombre de documents contenant l'attribut *A* dont la valeur est *v*.

4.4 Boosting

La technique de *boosting* utilise un comité de classifieurs $\phi_1, \ldots, \phi_k$ mais, contrairement aux autres méthodes voisines, l'ensemble de ces classifieurs est issu de la même méthode d'apprentissage, appelée l'apprenti *faible*, en passant par plusieurs itérations et en donnant à chaque fois plus de poids aux exemples incorrectement classifiés à l'itération précédente et en retenant pour les étapes subséquentes les classifieurs les plus prometteurs. Ces classifieurs sont entraînés séquentiellement, et non pas parallèlement, et ainsi un classifieur ϕ_i prend en compte la performance des classifieurs précédents $\phi_1, \ldots, \phi_{i-1}$ et essaie de se classifier correctement les exemples mal-classifiés par $\phi_1, \ldots, \phi_{i-1}$. Il est très important de noter que *Boosting* est une méthode générale visé améliorer la précision et la fidélité de n'importe quel algorithme d'apprentissage.

Notamment, pour entraîner un classifieur ϕ_t, chaque paire (d_j, c_i) est assignée un poids d'importance ${p^t}_{ij}$, (où p_{ij} représente le total de tous les poids assignés à l'ensemble des paires (d_j, c_i), $p_{ij} = \sum_{t=1}^{N} {p^t}_{ij}$). Ce poids représente le degré de la difficulté de prendre la décision de classification correcte face auquel les classifieurs $\phi_1, \ldots, \phi_{i-1}$ se sont heurtés. Ces poids seront exploités durant l'apprentissage de ϕ_t et seront ajustés, respectivement, pour résoudre les cas ayant un poids élevé. Ensuite le classifieur ϕ_i est appliqué sur les documents d'apprentissage et ainsi les poids ${p^t}_{ij}$ sont modifiés en ${p^{t+1}}_{ij}$: on diminue les poids des paires classifiées correctement tandis que les poids des paires mal-classifiées sont augmentés. Après avoir construit les k classifieurs, une combinaison linéaire des poids est utilisée pour former le comité final.

4.5 Évaluation des résultats du classifieur

Comme c'est le cas avec les systèmes de recherche d'information (*«information retrieval»*), nous nous basons sur l'expérience pour évaluer les classifieurs de textes, plutôt que de procéder analytiquement. Puisque toujours, la banque de documents est déjà étiquetée c'est-à-dire déjà classée et disponible, elle est divisée en deux ensembles: l'ensemble d'entraînement et l'ensemble de test. Dans ce qui suit, nous allons présenter les mesures de performance, que nous avons adoptées, et qui sont souvent utilisées dans la littérature pour des problèmes de classification. Pour mieux illustrer les différentes mesures qui vont suivre, on prend pour point de départ la table de contingence illustrée par le tableau 2.

Catégorie c_i	*Jugements de l'expert*	
	Document apartenant à la catégorie c_i	*Document n'apartenant pas à la catégorie* c_i

Jugements du Classifieur	*Document assigné à la catégorie par le classifieur*	VP_i	FP_i
	Document rejeté de la catégorie par le classifieur	FN_i	VN_i

Tableau 2 : *Table de contingence pour la catégorie* c_i

Où,
VP_i (Vrai Positif) est le nombre de documents correctement classés dans la catégorie $\boldsymbol{c_i}$,
FP_i (Faux Positif) est le nombre de documents incorrectement classés dans la catégorie $\boldsymbol{c_i}$,
VN_i (Vrai Négatif) est le nombre de documents correctement rejetés,
FN_i (Faux Négatif) est le nombre de documents incorrectement rejetés.

L'efficacité de la classification est mesurée suivant les deux notions classiques de la recherche d'information, notamment, la *précision* (π) et le *rappel* (ρ). En utilisant le tableau de contingence ci-dessus, les estimations de la précision et le rappel par rapport à une catégorie c_i sont calculés comme :

$$\hat{\pi}_i = \frac{VP_i}{VP_i + FP_i} \quad \text{et} \quad \hat{\rho}_i = \frac{VP_i}{VP_i + FN_i} \quad \text{[5] et [6]}$$

Ce qui est recherché, en fait, c'est une estimation globale et non une estimation pour chaque catégorie. Pour cela, dans cet article nous utilisons la méthode *macro-moyenne* en se basant sur le tableau de contingence globale (tableau 3) pour obtenir un score global.

L'ensemble des catégories $\boldsymbol{C = \{c_1, c_2, \ldots, c_{\lvert C\rvert}\}}$		*Jugements de l'expert*	
		Document apartenant à la catégorie $\boldsymbol{c_i}$	*Document n'apartenant pas à la catégorie* $\boldsymbol{c_i}$
Jugements du Classifieur	*Document assigné à la catégorie par le classifieur*	$VP = \sum_{i=1}^{\lvert C\rvert} VP_i$	$FP = \sum_{i=1}^{\lvert C\rvert} FP_i$
	Document rejeté de la catégorie par le classifieur	$FN = \sum_{i=1}^{\lvert C\rvert} FN_i$	$VN = \sum_{i=1}^{\lvert C\rvert} VN_i$

Tableau 3 : *Table de contingence globale pour la catégorie* c_i

La macro-moyenne calcule d'abord les scores pour chaque catégorie et fait ensuite une moyenne sur ces scores.

$$\hat{\pi}^M = \frac{\sum_{i=1}^{\lvert C\rvert} \hat{\pi}_i}{\lvert C\rvert} \quad \text{et} \quad \hat{\rho}^M = \frac{\sum_{i=1}^{M} \hat{\rho}_i}{\lvert C\rvert} \quad \text{[7] et [8]}$$

Où M indique la macro-moyenne.

Néanmoins, le mieux sera de calculer le *«seuil de rentabilité»*, c'est-à-dire le point où la précision et le rapel sont égaux. Pour cela, nous avons utilisé la mesure F_1 définie par:

$$F_1 = \frac{2\pi\rho}{\pi+\rho} \qquad [9]$$

C'est une fonction qui est maximisée quand la précision et le rappel sont proches. Nous cherchons généralement à l'optimiser lors de l'ajustement du seuil.

5. Description et conception du corpus

Notre corpus est composé de 1250 documents écrits en arabe, répartis entre cinq catégories préalablement choisies dont chacune contient 250 documents distincts. Les cinq catégories sont : *Politique*, *Economie*, *Médecine*, *Science et Technologie*, et *Sports*. Notre corpus a été conçu à partir des flux RSS diffusés par les agences de presse comme l'AFP, CNN Arabic, Al Jazeera, etc. Puisque chaque agence de presse dispose d'une mise en page propre à ses articles, nous étions obligés d'analyser l'agencement de chaque site pour identifier le début et la fin de leurs articles et puis fournir cette information à un logiciel développé en interne. Ce logiciel a pour fonction de recevoir chaque flux RSS et le traiter automatiquement pour extraire le texte de l'article associé à ce flux et le sauvegarder dans le répertoire correspondant à sa catégorie selon les étapes suivantes :

1. L'URL de l'article associé à chaque flux ainsi que sa catégorie sont extraits du flux.
2. L'outil ouvre automatiquement la page web de l'article et extrait son texte.
3. Toutes les balises et les caractères non-arabes sont supprimés du texte.
4. Les étapes élaborées dans la section 3.1 sont appliquées sur le texte résultant.
5. Enfin, le texte est sauvegardé dans un fichier texte dans le répertoire associé à sa catégorie.

6. Résultats

Pour effectuer nos expérimentations, nous avons utilisé le logiciel de fouille de données Weka[5]. L'algorithme utilisé pour la classification est *AdaBoost* and l'algorithme "boosté" est *C4.5*. Le nombre d'itérations *T* de boosting effectuée est de l'ordre de 10 et la mesure de validation appliquée est *la validation croisée stratifiée avec 10-plies*. La validation croisée est une technique statistique dans laquelle l'intégralité de l'espace d'apprentissage, c'est-à-dire les 1250 documents, est partitionné aléatoirement entre *k*-plies qui sont approximativement égaux (dans cet article *k*=10). Le partitionnement doit être le meilleur possible pour représenter un sous-ensemble de chaque classe dans

[5] http://www.cs.waikato.ac.nz/ml/weka/

chaque plie et c'est pourquoi on l'appelle *stratifiée*. Pour chacune des itérations *T* l'algorithme garde une partition à part et s'entraine sur les neufs restantes et son taux d'erreur est calculé en se basant sur la partition à part. Cette procédure est exécutée dix fois pour chaque itération *T* sur les différents dix sous-ensembles d'apprentissage (dont chacune a beaucoup en commun avec les neuf autres) et finalement le moyen des dix taux d'erreur est calculé pour l'itération *T*.

Nous avons mené une série d'expérimentations (dont les résultats d'un sous-ensemble sont présentés dans le tableau ci-dessous) pour trouver les meilleurs paramètres d'entrée qui produisent le meilleur résultat. Le paramètres d'entrée sont: le type d'attribut utilisé (les mots dans leurs formes d'origine, leurs lemmes, ou leurs racines), la technique de la RTV (GI, X^2, ou RG), et finalement le seuil de la technique de la RTV c'est-à-dire le nombre d'attributs gardés et utilisés pour former le vecteur global de l'espace d'apprentissage. Pour obtenir les racines et les lemmes des descripteurs nous nous sommes servi de l'analyseur morphologique de notre dictionnaire informatisé DIINAR[6].

Type d'attribut	**RTV?**	**Tech. de RTV**	**Nombre de termes**	**Précision -macro**	**Rapel-macro**	**F_1 -macro**
Forme Originale	Non		5087	0.7068	0.6296	0.6276
	Oui	GI	161	0.709	0.6088	0.6088
Lemmes	Non		4598	0.8396	0.8388	0.8376
	Oui	GI	161	0.8156	0.8152	0.8148
	Oui	X^2	161	0.8234	0.8232	0.8232
	Oui	RG	161	0.773	0.7416	0.753
Racines	Non		2943	0.8636	0.8624	0.863
	Oui	**GI**	**161**	**0.885**	**0.8848**	**0.8846**
	Oui	X^2	161	0.8816	0.88	0.8808
	Oui	RG	161	0.8436	0.8392	0.8402

Tableau 4 : *Un sous-ensemble des résultats des expérimentations menées*

La combinaison menant au meilleur résultat était celle utilisant les racines comme type d'attribut, le gain d'information comme technique de RTV avec un seuil de 161 attributs. Les taux de précision, rappel, et F_1 de cette combinaison sont présentés dans le tableau 5 ci-dessous.

Precision	**Recall**	**F-Measure**	**Category**
0.916	0.912	0.914	Economy
0.857	0.84	0.848	Medical
0.911	0.94	0.925	Politics
0.777	0.78	0.778	SciTech
0.964	0.952	0.958	Sports
0.885	.08848	0.8846	**Macro Avg**.

Tableau 5 *: Les taux de performance pour les cinq catégories de la combinaison gagnante.*

Nous avons essayé durant la série des expérimentations conduites d'ajuster le seuil des trois techniques de RTV en l'augmentant et le diminuant au-dessus et au-dessous de 161 mais cela n'a pas donné un meilleur résultat. GI était mieux que RG durant toutes les expérimentations ce qui signifie que les valeurs des attributs de l'espace d'apprentissage sont uniformément distribuées. L'utilisation du

[6] http://diinar.univ-lyon2.fr

module d'extraction des racines de notre dictionnaire informatisé DIINAR a amélioré la performance du classifieur comparée à l'utilisation des lemmes ou des mots dans leurs formes originales.

7. Conclusion

Cet article vise à adapter la méthode de la classification automatique des documents arabes en utilisant la technique de *Boosting*. Une étude comparative a été menée sur les trois mesures de RTV qui sont très bien connues et largement utilisées, notamment le *Gain d'Information (GI), Chi Carré (* χ^2*),* et *le Rapport de Gain (RG)*. Les résultats obtenus montrent que le GI avec un seuil de 161 produit la meilleure « fidélité » de classification avec une mesure moyenne de F1 égale à 0.8846 en se basant sur la validation croisée stratifiée avec 10-plies comme mesure d'évaluation. On a trouvé que *Boosting* a donné d'aussi bons résultats en fonction de la classification automatique des documents arabes que (Mesleh, 2007), qui a utilisé la méthode des machines à vecteurs de support (en anglais, *suport vector machines (SVM))* avec X^2 comme méthode de RTV et obtenu une mesure moyenne de F1 égale à 0.8811. On peut déduire, à priori, que *Boosting* est aussi performant que la méthode des SVM, qui est très bien connu d'être le meilleur classifieur de documents. Le futur travail sera d'augmenter la taille du corpus et de comparer la performance de SVM et de *Boosting* sur ce dernier. Pari ailleurs, on essayera de trouver les solutions nécessaires pour classer automatiquement les documents multilingues (c'est-à-dire les documents contenant en même temps plusieurs langues dans leurs textes) sans exclure le texte disponible dans les autres langues.

8. Bibliographie

M. Bardos*, Analyse Discriminante - Aplication au risque et scoring financier*, Paris : Dunod, 2001.

Cornuéjols A & Miclet L. *Aprentissage artificiel. Concepts et algorithmes*. 2001.

Forman, G. *An extensive empirical study of feature selection metrics for text classification*. J. Mach. Learn. Res. 3 , March 2003.

Govindarajan, M. *Text Mining Technique for Data Mining Aplication*. Proceedings of World Academy of Science, Engineering and Technology, volume 26, December 2007.

Hilbe, Joseph M., *Logistic Regression Models*. Chapman & Hall/CRC Press, 2009.

MacKay, David, *Information Theory, Inference, and Learning Algorithms*, 2003.

Mesleh, A. Suport vector machines based Arabic language text classification system: feature selection comparative study. In *Proceedings of the 12th WSEAS international Conference on Aplied Mathematics*, Cairo, Egypt, December 29 - 31 2007

Mitchell, T. M. *Machine Learning,* Computer Science, New York : McGraw-Hill, 1997

Peters, C. and Sheridan, P., Accès multilingue aux systèmes d'information, In 67th IFLA Council and General Conference, 2001

István Pilászy, Text Categorization and Suport Vector Machines. In the *Proceedings of the 6th International Symposium of Hungarian Researchers on Computational Intelligence*, 2005

Polity Y. L'organisation des connaissances en France : état des lieux. Communication aux *Journées d'étude du Chapitre français de l'ISKO, Lille*, 16-17 octobre 1997

Rish, I., An empirical study of the naive Bayes classifier*, IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, 2001

Salton, G. and McGill, M. J. *Introduction to Modern Information Retrieval*, McGraw-Hill, Inc. 1986

Saracevic, T., Information Science, *Journal of the American Society for Information Science* 50(12),1999, p. 1051-1063

Schapire, R. The Boosting Aproach to Machine Learning: An Overview. In *MSRI Workshop on Nonlinear Estimation and Classification*, 2002.

Sebastiani, F. Machine learning in automated text categorization, *ACM Computing Surveys,* 34(1): 1.47. 2002.

Shakhnarovish, D. and Indyk. *Nearest-Neighbor Methods in Learning and Vision*. The MIT Press, 2005

Sidhsom, S. *Plate-forme d'analyse morphosyntaxique pour l'indexation automatique et la recherché d'information: de l'écrit vers la gestion des connaissances.* Thèse de Doctorat à l'Université Claude Bernard Lyon1, France.2000.

Yang, Y. An evaluation of statistical aproaches to text categorization, *Information Retrieval*, 1 (1/2): 1999, p. 60-69