Représentation vectorielle de documents pour l'indexation de notices bibliographiques

Document vector embeddings for bibliographic records indexing

Morgane Marchand[1], Geoffroy Fouquier[1], Emmanuel Marchand[1], Guillaume Pitel[1]

[1] eXenSa, 41 rue Périer, Montrouge, France
morgane.marchand, geoffroy.fouquier, emmanuel.marchand, guillaume.pitel@exensa.com

RÉSUMÉ. Cet article présente la contribution d'eXenSa à l'édition 2016 du DÉfi Fouille de Textes (DEFT) dont la tâche consiste à indexer des documents scientifiques par des mots-clefs, préalablement sélectionnés par des professionnels. Le système proposé est purement statistique et combine une approche graphique et une approche sémantique. La première approche cherche dans le titre et le résumé du document des mots graphiquement proches des mots-clefs du thésaurus. La seconde approche attribue à un nouveau document des mots-clefs associés aux documents du corpus d'entraînement qui lui sont sémantiquement proches. Les deux approches utilisent des représentations vectorielles apprises en utilisant l'algorithme NC-ISC, un algorithme stochastique de factorisation de matrices. Notre approche a été classée première en terme de F-mesure sur deux des corpus de spécialité proposés et deuxième sur les deux autres.

ABSTRACT. This article presents the eXenSa contribution to the 2016 DEFT shared task. The proposed task consists in indexing bibliographic records with keywords chosen by professional indexers. We propose a statistical approach which combines graphical and semantic approaches. The first approach defines a document keywords as thesaurus terms graphically similar to terms contained in the title or the abstract of this document. The second approach assigns to document the keywords associated with semantically similar documents in training corpora. Both approaches use vector space models generated using NC-ISC, a stochastic matrix factorisation algorithm. Our system obtains the best F-score on half of the four test corpora and ranks second for the two others.

MOTS-CLÉS. Indexation, mots-clefs, espaces sémantiques, représentation vectorielle de mots.

KEYWORDS. Indexation, keywords, semantic spaces, word vector embedding.

1 Introduction

Avec la multiplication des données textuelles disponibles en ligne, l'indexation de document est devenue une thématique majeure afin de pouvoir fouiller de manière efficace dans les grandes bases de données. C'est également un prélude utile à de nombreuses autres tâches comme la classification de documents (Han *et al.*, 2007), la recherche d'information (Jones & Staveley, 1999) ou le résumé automatique (D'Avanzo & Magnini, 2005). L'indexation automatique est une tâche importante, que ce soit pour venir en aide à des indexeurs professionnels ou bien pour indexer des documents trop nombreux pour que des humains puissent tous les traiter. Elle peut prendre deux formes : l'indexation libre, qui est libre de choisir n'importe quel mot ou expression, et l'indexation contrôlée, qui doit choisir parmi un vocabulaire pré-établi. Dans la littérature, l'indexation libre est la plus étudiée. Cette catégorie regroupe des techniques très diverses(Hasan & Ng, 2014) : ordonnancement statistique(Salton *et al.*, 1975), classification binaire(Witten *et al.*, 1999) ou utilisation de graphes avec TextRank(Mihalcea & Tarau, 2004). Ces techniques sont souvent de type extractif, c'est à dire qu'elles sélectionnent comme mots-clefs des mots ou expressions présents dans les documents bien que rien n'interdise des reformulations.

L'indexation contrôlée quant à elle suppose l'existance préalable d'un vocabulaire spécifique au domaine du document à indexer. C'est souvent cette approche qui est de fait utilisée par les indexeurs professionnels. En effet, cela permet une homogénéisation des indexations, un des buts recherchés étant que chaque concept soit représenté par un et un seul mot-clef afin de faciliter la recherche dans la base de donnée. Comme un concept peut être exprimé de différentes façons, l'indexation contrôlée incite à des techniques au moins en partie attributives et non

seulement extractives même si cela n'est pas obligatoire. KEA++(Medelyan & Witten, 2006) est par exemple une méthode contrôlée extractive.

Plusieurs tâches d'évaluation se sont penchées sur le sujet et en particulier les ateliers DEfi Fouille de Texte. En 2012, la tâche proposée consistait à indexer des articles scientifiques par l'intermédiaire de mots-clefs d'auteurs (Paroubek *et al.*, 2012). Cette tâche mimait donc une indexation libre. L'équipe ayant obtenu les meilleurs scores a choisi une approche mixte utilisant des espaces sémantiques (El Ghali *et al.*, 2012).

L'édition 2016 de DEFT (Daille *et al.*, 2016) propose quant à elle de travailler sur l'indexation de documents scientifiques par des mots-clefs qui ont été proposés par des indexeurs professionnels. Cet article présente la contribution d'eXenSa à ce défi

Présentation de la tâche et des corpus

Les données mises à disposition par l'équipe de DEFT se composent de quatre corpus de spécialité : l'archéologie, la chimie, la linguistique et les sciences de l'information.

Chaque corpus est accompagné de son propre thésaurus. Les corpus sont des ensembles de notices bibliographiques issues des bases de données Pascal et Francis de l'INIST-CNRS. Les notices sont composées d'un titre, d'un résumé et d'une liste de mots-clefs attribués par un ingénieur documentaliste. Les mots-clefs sont soit issus d'un thésaurus de spécialité, soit ajoutés par l'ingénieur documentaliste en fonction de leur pertinence. Les organisateurs ont également mis à disposition les textes pré-traités au format TEI et texte.
Les corpus contiennent entre 706 et 782 notices dont 200 ont à chaque fois été sélectionnées au hasard pour constituer le corpus de test. Le nombre de mots-clefs par notice varie entre 7 et 30 mots-clefs. La part des mots-clefs sélectionnés directement dans le texte du résumé est variable suivant les corpus (de 44 à 60 %).

Pour l'évaluation, les mesures qui ont été retenues par les organisateurs du défi sont les mesures de précision, de rappel et de F-mesure (Manning *et al.*, 1999), calculées avec une macro-moyenne. Ces mesures avaient déjà été utilisées auparavent pour la tâche 5 de la campagne Sem-Eval-2010 qui s'intéressait à l'extraction automatique d'expressions clefs libres dans des articles scientifiques annotés par leurs auteurs et leurs lecteurs (Kim *et al.*, 2010) :

$$P = 100\frac{\Sigma_{doc}P(doc)}{N}$$

$$R = 100\frac{\Sigma_{doc}R(doc)}{N}$$

$$F = 100\frac{\Sigma_{doc}F(doc)}{N}$$

avec $P(doc)$, $R(doc)$ et $F(doc)$ respectivement la précision, le rappel et la F1-mesure obtenues pour un document et N, le nombre total de documents.

Notre approche utilise des représentations vectorielles des mots et des documents, factorisées afin de gagner en efficacité. Nous utilisons à la fois une méthode graphique extractive à partir d'une matrice mots-graphèmes, présentée dans la partie 3 et une méthode sémantique attributive à partir d'une matrice document-mots, présentée dans la partie 4. Le couplage de ces deux méthodes est décrit dans la partie 5 avant de conclure en partie 6 par une discussion des résultats.

2 État de l'art

L'indexation de documents, automatique (Salton *et al.*, 1975) ou non (Lancaster *et al.*, 1991), est un domaine de recherche fourni. Les termes clefs sont en effet très utiles pour faciliter la recherche dans de grandes bases de données (Medelyan & Witten, 2008) ou pour la construction automatique de résumés (Wan *et al.*, 2007; Litvak &

Last, 2008; Boudin & Morin, 2013). Pour obtenir les meilleurs résultats, il est souvent pertinent de combiner une approche automatique avec une approche manuelle (Savoy, 2005).
La plupart des techniques d'indexation automatique comprennent deux phases (Paroubek *et al.*, 2012) : une première de présélection des candidats et une seconde de réordonnancement et de sélection finale.

Dans le cadre d'une indexation s'appuyant sur un lexique, la présélection peut par exemple s'effectuer par projection des termes du lexique de spécialité sur le texte à indexer, avec la suppression des mots vides et la prise en compte des lemmes. Une extension incluant des variantes peut être effectuée en prenant en compte les coordinations, les modifieurs et les permutations (Jacquemin, 1997; Hamon, 2012) ou bien en apprenant automatiquement des règles de dérivation par *bootstrapping* à partir d'un petit nombre d'exemples (Moreau *et al.*, 2007; Claveau & Raymond, 2012). Enfin, cette projection peut s'effectuer à partir de représentations sémantiques vectorielles des termes clefs et des documents. Il s'agit alors pour indexer un document de sélectionner les termes ayant une représentation vectorielle similaire à la sienne (El Ghali *et al.*, 2012).
Dans le cadre d'une indexation libre, la plupart des méthodes automatiques sélectionnent au préalable les portions de texte qui pourraient servir de termes clefs. Pour effectuer cette sélection, une méthode possible est d'utiliser des patrons syntaxiques et des règles de sélection manuelles concernant les parties du discours (Tonelli *et al.*, 2012), ou bien de sélectionner des n-grammes de taille prédéfinie ne commençant ni ne terminant par un mot vide (Boudin *et al.*, 2012), ou encore de détecter les chaînes répétées maximales de mots (Doualan *et al.*, 2012).

Les termes présélectionnés sont ensuite très souvent pondérés par diverses mesures statistiques (Paroubek *et al.*, 2012), telles que la fréquence d'apparition dans un texte ou son résumé (*term frequency - tf*) et la fréquence d'apparition dans les documents du corpus (*document frequency - df*), ainsi que leur composés tf-idf ou okapi-bm25 (Ding *et al.*, 2011; Claveau, 2012). La position de la première occurrence du candidat est aussi fréquemment utilisée, ainsi que sa longueur ou bien son score de saillance dans son arbre de dépendances (Boudin *et al.*, 2012). Les longs termes clefs peuvent également être jugés plus informatifs et à privilégier (Barker & Cornacchia, 2000).
Afin d'améliorer la couverture et la cohérence de l'indexation, les termes présélectionnés peuvent être regroupés en fonction de leur lien sémantique (Matsuo & Ishizuka, 2004; Liu *et al.*, 2009). Un seul candidat par regroupement est alors retenu.
D'autres travaux encore, utilisent des algorithmes de graphes (Mihalcea & Tarau, 2004; Wan & Xiao, 2008; Ahat *et al.*, 2012) ou des réseaux bayésiens (Witten *et al.*, 1999; El Ghali *et al.*, 2012) afin de prendre en compte dans l'apprentissage des informations extérieures aux textes, comme par exemple le nom de la revue de publication d'un article ou ses documents voisins.

2.1. Indexations contrôlées extractives et attributives

La campagne DEFT 2016 présente une tâche d'indexation contrôlée. Le contrôle peut être exercé de deux façons. Soit on dispose de candidats extraits du texte et on les filtre et les reformule à l'aide des mots-clefs du thésaurus, il s'agit alors d'indexation contrôlée extractive. Soit on attribue à un texte des mots-clefs du thésaurus sans se préoccuper de leur présence dans le texte mais en se focalisant plutôt sur une similarité de thématique. Il s'agit alors d'indexation contrôlée attributive. La méthode KEA++(Medelyan & Witten, 2006) est un exemple de méthode contrôlée extractive. Dans un premier temps, les mots ou expressions du thésaurus présents dans le document sont retenus comme mots-clefs possibles. Puis les candidats sont départagés en utilisant un classifieur naïf bayésien s'appuyant sur trois traits statistiques : la position de la première apparition du mot candidat (Witten *et al.*, 1999), son tf-idf, indice numérique mixant sa fréquence locale et sa distribution en corpus, ainsi que le nombre de relations à l'intérieur du thésaurus si celui-ci en dispose.
Dans une logique d'approche contrôlée attributive, on recherche les mots-clefs les plus proches des documents d'un point de vue thématique. Des représentations vectorielles dans des espaces sémantiques compatibles des documents et des mots-clefs proposés peuvent alors être utilisées (El Ghali *et al.*, 2012; Chartier *et al.*, 2016a).
Notre approche utilise quant à elle des représentations vectorielles sémantiques denses de documents et de mots considérés comme des documents de graphèmes.

2.2. Représentations vectorielles denses

Une façon simple d'associer un vecteur à un mot lorsque l'on dispose d'un corpus, est par exemple de construire une matrice de cooccurrences, où l'on compte le nombre de fois où deux mots apparaissent dans une fenêtre de taille donnée. Les vecteurs obtenus ainsi ont pour dimension la taille du vocabulaire. Ils sont donc creux et de grande dimension. Il est également possible de construire des matrices document-mots où l'on compte le nombre de fois où un mot apparait dans un document. Les vecteurs représentant les documents ont donc une dimension égale à la taille du vocabulaire et les vecteurs de mots une dimension égale à la taille du corpus. Plusieurs pondérations peuvent être appliquées à ces scores bruts, les plus classiques étant la PMI, mesure d'association entre deux variables aléatoires, pour les matrices de cooccurrence et le tf-idf pour les matrices document-mots.

Un inconvénient de ces représentations creuses est que plus le corpus et le vocabulaire augmentent, plus les matrices deviennent grandes et coûteuses à manipuler. Plusieurs stratégies existent pour obtenir des représentations vectorielles denses des mots. Ces représentations sont de taille fixe, souvent bien plus petite que la taille du vocabulaire, et les valeurs que prennent les différents traits sont des réels. Ces représentations denses occupent une plus faible place en mémoire et permettent des traitements de données plus rapides. De plus, il a été montré que ces représentations vectorielles denses sont efficaces pour capturer les régularités linguistiques d'un corpus, permettant d'appréhender les questions de similarités syntaxiques et sémantiques par de l'arithmétique de vecteurs (Mikolov *et al.*, 2013c; Levy *et al.*, 2015).
Certaines techniques, comme word2vec (Mikolov *et al.*, 2013b,a), créent directement ces représentations, en entraînant un réseau de neurones sur un corpus et en assignant comme représentation à un mot les valeurs prises par la dernière couche cachée du modèle. Un autre type d'approche consiste à factoriser les matrices creuses précédemment construites. C'est le cas de la LSI (Latent Semantic Indexing) (Deerwester *et al.*, 1990), spécifiquement créée pour la recherche d'information, qui factorise une matrice document-mots pondérée par tf-idf. La matrice est alors réduite à la taille souhaitée à l'aide d'une décomposition en valeurs singulières. LSA (Latent Semantic Analysis) (Landauer & Dumais, 1997) est l'utilisation de la technique de réduction de dimension de la LSI sur un autre type de matrices (par exemple une matrice de cooccurrence pondérée par PMI).

2.3. L'algorithme NC-ISC

L'algorithme de factorisation que nous avons employé pour ce défi est l'algorithme NC-ISC, ou *Norm Constrained Iterative Semantic Characterization* [1]. Il s'agit d'un algorithme stochastique de factorisation de matrices de la famille de RandNLA (Drineas & Mahoney, 2016). Les algorithmes de projection aléatoire sont en effet bien adaptés pour fonctionner sur des architectures parallèles et sont performants sur de grandes dimensions (Halko *et al.*, 2011). NC-ISC permet d'apprendre des représentations vectorielles sur des tâches non supervisées ou semi-supervisées. Ces vecteurs peuvent être utilisés directement dans des recherches de plus proches voisins ou bien servir d'étape de prétraitement pour d'autres tâches d'apprentissage.

A l'instar de la LSA (Landauer & Dumais, 1997), NC-ISC effectue une réduction de dimension en réalisant une factorisation d'une grande matrice creuse en trois matrices denses : une matrice contenant des représentations vectorielles pour la première dimension de la matrice, une matrice contenant les valeurs propres des dimensions latentes et une matrice contenant des représentations vectorielles pour la seconde dimension de la matrice. La principale différence provient des étapes de normalisation intermédiaires qui permettent de contraindre les normes des vecteurs et l'expension des valeurs propres. Ainsi les poids des mots fréquents et des mots non fréquents sont au final plus similaires.

De plus, la factorisation QR utilisée dans la LSA demande du temps de calcul. NC-ISC introduit une factorisation itérative qui accélère le temps de calcul.

Pour illustrer ce gain de temps, nous avons effectué une comparaison avec différents algorithmes, dont les résultats sont présentés au tableau 2.1. Il s'agit de créer des représentations vectorielles de dimension 500 à partir d'une matrice de cooccurrences pondérée par la PMI. Nous avons effectué nos calculs sur le corpus utilisé par

1. L'algorithme NC-ISC est développé par la société eXenSa dans son moteur eXenGine (http ://www.exensa.com/about-us/). Une démonstration d'eXenGine analysant les données de wikipédia anglais est disponible à l'adresse http ://www.wikinsights.org.

(Levy *et al.*, 2015). Il s'agit d'un export de wikipédia dans sa version anglaise datant d'août 2013, tokenisé et séparé en phrases. Les tokens apparaissant moins que 100 fois sont supprimés. Au final, le corpus contient donc 77,5 million de phrases pour un vocabulaire de 189 533 tokens. Les cooccurrences sont prises en compte avec une fenêtre de taille 2.
Pour la SVD, nous avons utilisé l'implémentation de scikit-learn [2], modifié afin de paralléliser au maximum le processus. Le temps présenté, pour SVD et NC-ISC, est celui nécessaire à la factorisation de la matrice de PMI pré-calculée. Word2vec, dans son implémentation *skip-gram negative sampling (SGNS)* (Mikolov *et al.*, 2013b), est une méthode état de l'art pour obtenir des représentations vectorielles sémantiques de mots fonctionnant très bien pour des tâches de détection d'analogie. (Levy & Goldberg, 2014b) ont montré que les résultats obtenus par SGNS pouvaient s'apparenter à la factorisation d'une matrice de PMI positive décalé d'une constante. Nous présentons donc également son temps de calcul à partir de couples mot-contexte pré-calculés (Levy & Goldberg, 2014a).

Méthode	Temps de calcul (s)
SVD	2078
SGNS	1659
NC-ISC (10 itérations)	115

Tableau 2.1. *Temps de calcul en secondes pour le calcul de représentations vectorielles de dimension 500 à partir d'une matrice de cooccurrences pondérée par PMI sur un corpus issu de wikipédia.*

Nous observons donc que le temps de calcul pour NC-ISC est bien plus rapide. De plus, dans des travaux précédents, nos représentations vectorielles calculées à l'aide de NC-ISC donnaient des résultats similaires à ceux de word2vec dans des tâches de recherche de similarité [3]. Nos performances sont légèrement plus faible pour la similarité (*similarity*) mais légèrement plus élevées pour les mots liés (*relatedness*).

Un autre avantage de NC-ISC, est le fait que des connaissances a priori peuvent facilement être introduites lors des itérations sur chacune des dimensions, sous forme de vecteurs pré-calculés qui sont concaténés aux matrices en cours de calcul. Cela permet par exemple d'infléchir le calcul afin que deux objets appartenant à une même catégorie (ou deux mots utilisés dans un même contexte) aient au final des représentations approchantes.

Pour ce défi, deux types de matrices ont été utilisés : une matrice mot-graphèmes pour construire des représentations vectorielles graphiques et une matrice document-mots pour construire des représentations vectorielles sémantiques.

3 Modèle graphique

Notre première stratégie pour attribuer des mots-clefs à un document est une stratégie extractive. Nous avons émis une hypothèse formulée ainsi : le titre et le résumé d'un article permettent de connaître l'objet de celui-ci. De cette hypothèse découle une règle de sélection des mots-clefs à partir du thésaurus thématique correspondant : un mot-clef est un terme du thésaurus dont tous les mots pleins qui le composent apparaissent dans le titre ou le résumé d'un article (recherche exacte). Nous nous limitons dans notre recherche aux mots-clefs des thésaurus qui ont été utilisés dans au moins un document du corpus d'entraînement.

	Précision	Rappel	F-mesure
Archéologie	46,4	17,9	24,5
Sciences de l'information	18,4	8,5	10,8
Chimie	26,9	6,4	9,6
Linguistique	17,6	6,9	9,5

Tableau 3.2. *Scores obtenus par recherche exacte de mots-clefs dans les titres et résumés.*

2. http ://scikit-learn.org/stable/index.html
3. http ://alfonseca.org/eng/research/wordsim353.html

Le tableau 3.2 montre que les scores obtenus en recherchant les mots-clefs de manière exacte sont assez faibles, surtout en ce qui concerne le rappel. En effet, les mots ou expressions-clefs apparaissant dans les titres et les résumés n'apparaissent pas forcément sous une forme référencée dans le thésaurus. Or, comme nous n'utilisons pas de prétraitement linguistique comme la lemmatisation, les mots au pluriel (ou accordés au féminin) par exemple sont donc considérés comme des mots différents d'une version au singulier (resp. au masculin).

3.1. Recherche approchée : lemmatisation, racinisation et graphèmes

3.1..1. Lemmatisation

Dans le cas de langues pour lequelles des ressources linguistiques sont disponibles, il est donc plus judicieux de réaliser l'extraction de mots-clefs sur des lemmes. Pour cette évaluation, nous avons utilisé le lemmatiseur fournit dans treetaggerwrapper [4]. Comme le montre le tableau 3.3, l'utilisation de la lemmatisation permet un gain significatif en terme de F-mesure.

	Précision	Rappel	F-mesure
Archéologie	48,1	33,0	37,4
Sciences de l'information	17,7	11,1	12,9
Chimie	25,5	10,6	13,9
Linguistique	23,3	14,1	16,8

Tableau 3.3. *Scores obtenus par recherche exacte de mots-clefs dans les titres et résumés après lemmatisation.*

Les quatre corpus ne se comportent cependant pas de la même façon. La précision est bien meilleure pour chimie et linguistique alors qu'elle reste stable, voire décroit pour les deux autres corpus. Le gain en rappel est toujours présent, surtout marqué pour les corpus sciences de l'information et linguistique. Au final, les gains en F-mesure vont de +2,1 à +12,9.
Le gain en précision s'explique si les mots-clefs ajoutés à la proposition sont en grande majorité des mots-clefs corrects, c'est à dire si les mots-clefs sont souvent utilisés sous leur forme déclinée dans les résumés.

Une autre possibilité est la racinisation. Nous l'avons effectuée avec le module python nltk. Par rapport aux résultats sur le corpus lemmatisé, la précision diminue et le rappel augmente. C'est en effet logique étant donné que l'évaluation se fait sur la présence ou l'absence exacte des mots-clefs sans prendre en compte ceux qui sont de la même famille, qui peuvent se confondre suite à la racinisation.

	Précision	Rappel	F-mesure
Archéologie	40,1	44,7	40,4
Sciences de l'information	15,5	17,3	15,7
Chimie	22,6	13,4	15,3
Linguistique	16,7	19,2	16,8

Tableau 3.4. *Scores obtenus par recherche exacte de mots-lefs dans les titres et résumés après racinisation.*

3.1..2. Représentation par graphèmes

Lorsque les ressources linguistiques sont absentes, une possibilité pour rapprocher des mots partageant la même racine est de s'intéresser aux graphèmes le composant.
Une approche similaire a été utilisée par une autre équipe participante, utilisant les graphèmes pondérés par une mesure du χ^2 pour représenter les mots-clefs et des graphèmes pondérés par un tf-idf pour représenter les documents(Chartier *et al.*, 2016a).
Cette stratégie se rapproche également des travaux de (Bojanowski *et al.*, 2016). Ils utilisent la même approche

4. https ://pypi.python.org/pypi/treetaggerwrapper ; http ://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger/

que word2vec mais utilisent un vocabulaire étendu composé des mots complets ainsi que des graphèmes. Les graphèmes sont utilisés pour la représentation des mots peu fréquents.

Pour que les différentes versions d'un mot puissent être rapprochées, il est possible de représenter un mot en sac de graphèmes. La taille des graphèmes ou l'utilisation de marqueurs de début et de fin peut influencer les rapprochements entre mots.

Par exemple, si l'on utilise une décomposition en graphèmes de 2 à 5 lettres ordonnées avec des marqueurs de début de mot mais sans marqueur de fin, le mot « laches » sera représenté de la manière suivante :

```
[::s::l; la; ac; ch; he; es;
 ::s::la; lac; ach; che; hes;
 ::s::lac; lach; ache; ches;
 ::s::lach; lache; aches]
```

Ces représentations sont concaténées en une matrice mot-graphèmes dont on réduit la dimension à l'aide de NC-ISC afin d'obtenir des vecteurs denses de représentation graphique des mots. La recherche approchée de mots-clefs dans les textes se déroule alors comme suit : un mot-clef est retenu pour un article si chacun des mots pleins qui le composent a une similarité dépassant un certain seuil, fixé empiriquement, avec l'un des mots du titre ou du résumé. La similarité utilisée est la mesure du cosinus entre les vecteurs denses de représentation. Ainsi, les vecteurs de « lexique » et « lexiques » ont une similarité très élevée entre eux, mais sont également proches du vecteur représentant le mot « lexical ». Le tableau 3.5 donne quelques exemples de mots-clefs retrouvés de cette façon.

mots du texte	mot-clef	distance
citations	citation	0,993
mécanismes discursifs [...] stratégie de réduction	stratégie discursive	0,984
visée argumentative	argumentation	0,977
développement langagier	développement du langage	0,963
bilingue	bilinguisme	0,957

Tableau 3.5. *Exemple de mots-clefs correctement identifiés avec leur distance graphique aux mots présents dans le texte.*

Taille des graphèmes

Nous avons testé la représentation graphique à l'aide de 2-grammes de lettres uniquement, de 5-grammes de lettre uniquement ou bien des 2- aux 5-grammes de lettres.

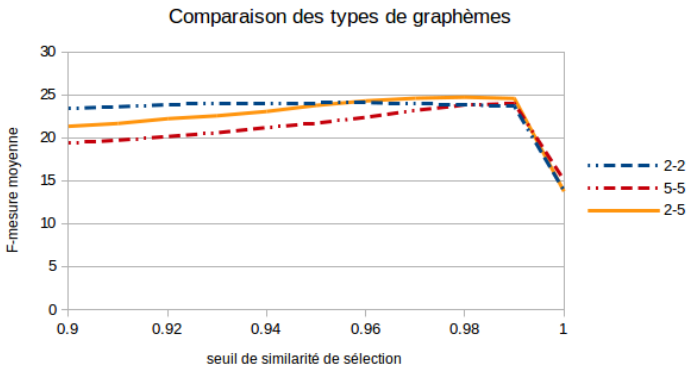

FIGURE *3.1.* *F-mesure moyenne en fonction du seuil de sélection des graphèmes et de leur type.*

La figure 3.1 présente la F-mesure moyenne obtenue sur les quatre corpus avec des graphèmes 2-2, 5-5 ou 2-5 en fonction du seuil de sélection utilisé pour déterminer si deux mots sont ou non similaires. Un seuil de 1 correspond

au cas de sélection exacte développé plus haut et qui a de faibles performances. Quel que soit le type de graphème et le seuil de sélection employés, l'utilisation d'une recherche approchée augmente donc grandement les résultats.

On constate que la majorité de l'information est portée par les 2-grammes, qui sont performants avec des seuils bien plus faibles que les 5-grammes, même si les courbes finissent par se rejoindre, voire se croiser pour le corpus sciences de l'information.
La combinaison des différents graphèmes donne de meilleurs résultats que les deux types de graphèmes seuls pour des seuils élevés pour les corpus chimie, linguistique et sciences de l'information. Ce comportement se reflète dans la F-mesure moyenne. Afin d'adopter une stratégie qui s'approche de l'optimal pour tous les corpus nous avons retenu la combinaison de graphèmes 2-5 avec un seuil de 0,98.

Marqueurs de début et de fin

Une autre variante possible des graphèmes est la présence ou l'absence de marqueurs de début et de fin. Ils ont pour effet de donner plus de poids au début ou à la fin d'un mot. La présence de marqueur de fin serait donc un handicap lorsque l'on cherche à rapprocher plusieurs dérivés d'un même mot. Cependant, en ce qui concerne les quatre corpus utilisés au cours de DEFT, la présence ou l'absence de ces marqueurs n'a pas fait montre d'influence significative. Dans d'autres études, l'absence de marqueur de fin s'étant néanmoins révélée utile au rapprochement des mots de la même famille, c'est cette version qui a été retenue.

Dimension des vecteurs graphiques

Comme nos vecteurs sont obtenus par factorisation de matrice, la taille de réduction choisie peut avoir de l'influence sur le résultat. Cependant, la figure 3.2 montre que ce paramètre ne semble pas entraîner une grande variabilité.

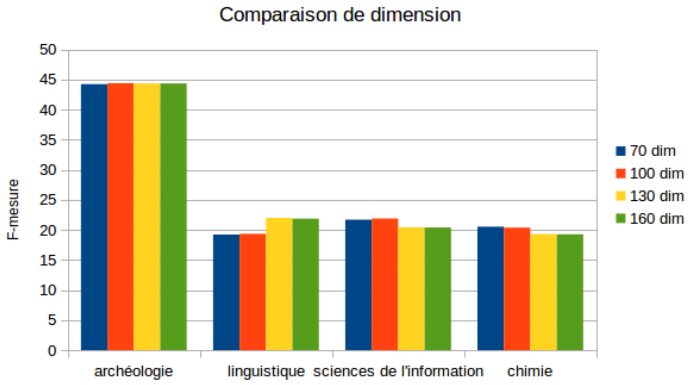

FIGURE *3.2.* *F-mesure en fonction de la dimension de réduction de la matrice graphique.*

Résultats

Au final, les vecteurs graphiques s'étant révélés en moyenne les plus efficaces sont les vecteurs de graphèmes 2-5, avec marqueur de début et sans marqueurs de fin, avec une dimension de réduction de 130 et un seuil de sélection de 0,98. Les résultats obtenus sont rapportés au tableau 3.6.

Cette approche graphique approchée permet une forte augmentation du rappel ainsi qu'une faible baisse de la précision (en moyenne) par rapport à l'approche exacte. L'amélioration en terme de F-mesure va de $9,8$ à $19,9$ par rapport à l'approche exacte.

Le tableau 3.7, quant à lui, rassemble les résultats obtenus par les différentes approches graphiques. L'utilisation des graphèmes permet d'obtenir un rappel bien plus élevé qu'en utilisant les lemmes ou les racines. Cela est sans

	Précision	Rappel	F-mesure
Archéologie	43,5	50,0	44,4
Linguistique	20,6	26,0	22,0
Sciences de l'information	19,7	23,6	20,4
Chimie	25,5	17,8	19,3

Tableau 3.6. *Scores obtenus par recherche approchée de mots-clefs dans les titres et résumés en utilisant des représentations vectorielles graphiques et un seuil de similarité de sélection de* $0,98$.

		Précision	Rappel	F-mesure
Archéologie	exact	46,4	17,9	24,5
	graphèmes	43,5	**50,0**	**44,4**
	lemmes	**48,1**	33,0	37,4
	racines	40,1	44,7	40,4
Linguistique	exact	17,6	6,9	9,5
	graphèmes	20,6	**26,0**	**22,0**
	lemmes	**23,3**	14,1	16,8
	racines	16,7	19,2	16,8
Sciences de l'information	exact	18,4	8,5	10,8
	graphèmes	**19,7**	**23,6**	**20,4**
	lemmes	17,7	11,1	12,9
	racines	15,5	17,3	15,7
Chimie	exact	**26,9**	6,4	9,6
	graphèmes	25,5	**17,8**	**19,3**
	lemmes	25,5	10,6	13,9
	racines	22,6	13,4	15,3

Tableau 3.7. *Comparaison des approches graphiques.*

doute dû au fait que le modèle graphique peut rapprocher des mots plus éloignés ou même, comportant des fautes d'orthographe. De plus, la précision obtenue est toujours supérieure à celle obtenue avec la racinisation. Utiliser une représentation en graphèmes lorsque l'on ne dispose pas de ressources linguistiques adaptées à la langue traitée est donc une approche tout à fait pertinente, et même recommandable si on s'intéresse au rappel.

3.2. Discussion

Notre modèle graphique est une technique d'indexation par extraction contrôlée. Trois autres équipes participant à DEFT 2016 ont également utilisé une approche extractive. Pour (Bougouin *et al.*, 2016), leur approche purement extractive utilisant des graphes donne de moins bons résultats que leur version purement attributive. Les deux autres équipes ont cependant des résultats se rapprochant de nos observations. (Hamon, 2016) réalise une acquisition terminologique sur les documents lemmatisés en prenant en compte des variantes morpho-syntaxiques des termes du thésaurus. (Buscaldi & Zargayouna, 2016) quant à eux indexent au préalable le thésaurus pour assigner à chaque concept des représentations lexicales, ici des trigrammes, qui permettront de repérer les concepts présents, même partiellement, dans les textes. Leurs résultats sont cohérents avec les nôtres, ces méthodes d'extraction se révélant particulièrement précises sur le corpus d'archéologie, pour lequel beaucoup de termes clefs choisis par les indexeurs professionnels sont présents dans les textes, et dans une moindre mesure sur celui de chimie.

4 Modèle sémantique

L'approche précédente, par construction, ne permet d'attribuer à une notice que les mots-clefs dont des variantes sont présentes dans le titre ou le résumé. Cependant, certains mots-clefs pertinents, notamment des mots désignant des domaines ou phénomènes plus génériques, ne seront pas présents directement dans le texte. Par exemple

un texte parlant de la synthèse de composés aromatiques n'utilisera pas nécessairement l'expression "synthèse chimique" dans son résumé alors qu'il s'agit d'un mot-clef pertinent.

Ce type de comportement peut être émulé à l'aide d'une logique attributive peut être employée. Pour traiter cet aspect du problème, nous avons choisi d'utiliser un espace vectoriel sémantique construit à partir d'une matrice document-mots. Les espaces sémantiques vectoriels représentent des mots et des documents sous forme de vecteurs dont la distance est facilement mesurable. Ils permettent donc de rapprocher des textes qui ne partagent pas ou peu de mots en commun si leurs vecteurs représentatifs sont proches.

4.1. Matrice document-mots

Pour construire notre modèle sémantique, nous avons utilisé une matrice document-mots représentant les documents des corpus en sacs de mots d'uni-, bi- et tri-grammes, sélectionnés selon un critère d'information mutuelle. Les valeurs de cette matrice sont pondérées par le score tf-idf des n-grammes, mesure de pondération classiquement utilisée (Sparck Jones, 1972). La dimension de la matrice est ensuite réduite à l'aide de l'algorithme NC-ISC.

Lorsque l'on dispose de représentations vectorielles pour les résumés et les mots-clefs des thésaurus, il est possible d'attribuer des mots-clefs aux notices en comparant directement leurs vecteurs via la distance du cosinus pour ordonner les candidats. C'est une approche de ce type qu'ont retenue (Chartier *et al.*, 2016a) en utilisant des matrices document-graphèmes et mot-clef-graphèmes, pondérées respectivement par okapi et χ^2. Cependant, en ce qui concerne les matrices que nous avons utilisées, d'après nos expériences, cette approche ne donne pas les meilleurs résultats.
Nous avons donc adopté une approche qui s'inspire de la méthode de classification par plus proches voisins. Pour un document donné, nous déterminons quels sont les documents les plus similaires puis, connaissant leur mots-clefs respectifs, nous en déduisons les mots-clefs liés à ce nouveau document. Il est en effet possible de considérer la tâche d'indexation attributive comme une tâche de classification multi-classes, multi-étiquettes (Zhang & Zhou, 2014; Partalas *et al.*, 2013).

4.2. Création des vecteurs des documents du corpus de test

Après apprentissage, chaque document du corpus d'entraînement dispose d'une représentation vectorielle et chaque mot apparaissant dans le corpus dispose également d'une telle représentation. Pour qu'il soit possible de comparer les documents du corpus de test aux documents du corpus d'apprentissage, chaque document du corpus de test doit être représenté par un vecteur compatible. La représentation d'un document est construite à partir des vecteurs des mots qu'il contient. C'est une approche semblable qui est appliquée dans doc2vec, l'extension aux documents de word2vec(Le & Mikolov, 2014). Cette représentation est donc obtenue en sommant membre à membre les vecteurs représentant chacun des mots contenus dans le document. Il est ensuite nécessaire d'ôter de ce vecteur les valeurs propres de la matrice document-mots réduite, calulées sur le corpus d'entraînement par NC-ISC, afin d'obtenir une représentation comparable aux vecteurs des documents d'entraînement. Pour estimer la similarité entre deux documents dans cet article, nous utilisons le complément à 1 de la distance cosinus entre leurs représentations vectorielles.

Dans le cadre du défi, nous disposions du contenu des documents test. Nous aurions donc pu apprendre les représentations des mots et des documents sur un corpus élargi constitué à la fois des documents d'entraînement et des documents de test. Il aurait alors été inutile de construire *a posteriori* une représentation des documents test. De plus, entraînées sur plus de données, les représentations vectorielles des documents auraient sans doute été plus précises. Nous avons cependant choisi de rester dans un cas d'utilisation où, quand de nouveaux documents se présentent, il est possible de trouver leurs mots-clefs sans avoir besoin de recalculer la représentation. Il s'agit du cas d'usage où une base documentaire est enrichie en continu avec de nouveaux articles. Il n'est alors plus nécessaire de disposer de l'algorithme pour réaliser l'indexation, ce qui est pratique pour un centre documentaire. Les vecteurs pré-calculés gardés en mémoire suffisent.

4.3. Calcul des plus proches voisins

Après le calcul des représentations de chaque document du corpus de test, l'attribution des mots-clefs s'effectue selon une approche kNN de la manière suivante :

— identification des K plus proches voisins parmi les documents du corpus d'entrainement de la spécialité associée.
— suppression des documents voisins ayant une similarité inférieure à un seuil S avec le document test.
— remplacement des documents voisins par leurs mots-clefs associés, pondérés par leur similarité respective avec le document test. Si un même mot-clef est lié à plusieurs documents voisins du document test, alors les scores de similarité sont additionnés.
— prise en compte de la popularité des mots-clefs candidats dans le corpus d'entraînement : leur score de similarité est multiplié par le logarithme du nombre d'apparitions du mot-clef dans le corpus d'entraînement.

Cette méthode est à rapprocher des travaux de (Ishii *et al.*, 2006a,b). Ils utilisent la méthode de classification par kNN après avoir effectué une LSA et obtiennent des résultats se rapprochant des SVM.

La somme des scores de similarité en cas d'apparition multiple d'un même mot-clef favorise les mots-clefs les plus utilisés dans la base d'apprentissage, étant donné que plus de voisins sont susceptibles de les partager. Néanmoins un mot-clef rare utilisé par un document très similaire au texte à classer pourra être sélectionné.

4.4. Etude des paramètres

4.4..1. Paramètres du kNN

Les mots-clefs finalement sélectionnés dépendent du résultat de la recherche des K plus proches voisins. La figure 4.3 montre néanmoins que les résultats finaux sont assez stables par rapport aux paramètres de sélection des documents voisins (limite en nombre de voisins K et en similarité S). On observe toutefois qu'il faut un seuil de similarité assez bas au risque d'obtenir des socuments sans aucun plus proche voisin. C'est ce phénomène qui entraîne une baisse de la courbe lorsque le seuil de similarité augmente. Cette baisse se poursuit ensuite jusqu'à ce que plus aucun document ne passe le seuil.

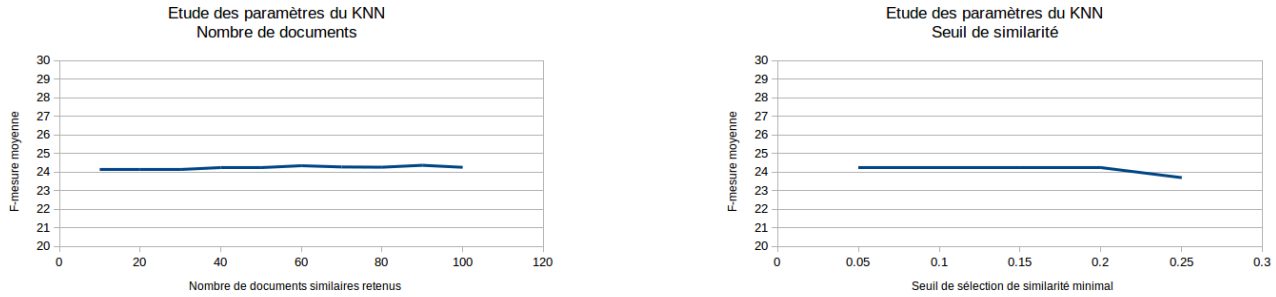

FIGURE 4.3. *Etude des paramètres du kNN.*

4.4..2. Seuil de sélection final

Une fois la liste de mots-clefs obtenue et ordonnée, il reste à définir le seuil de sélection final de ces mots-clefs. La figure 4.4 présente l'évolution de la F-mesure pour les quatre corpus de test.

Avec le modèle graphique seul, le seuil de sélection à adopter est très marqué. Il s'agit d'un seuil minimum de 0,2 de similarité cosinus entre deux textes voisins pour le sélectionner et utiliser ses mots-clefs.
Plus le seuil est élevé et plus la précision du résultat est élevée. Cependant, le rappel est alors faible, en raison notamment du grand nombre de mots-clefs attribué à chaque document. Pour obtenir une F-mesure satisfaisante, la valeur du seuil doit donc être assez basse.

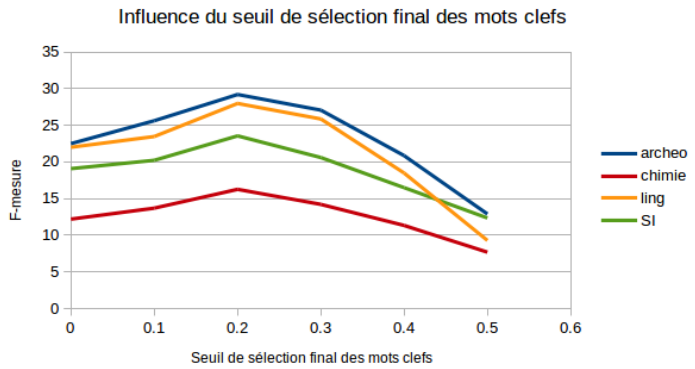

FIGURE 4.4. *Influence du seuil de sélection final des mots-clefs.*

4.5. Utilisation d'apriori lors de la factorisation

Afin de forcer la convergence des vecteurs que l'on sait devoir être proches, il est possible, au cours de la réduction itérative, de concaténer la matrice obtenue à une itération avec une matrice précalculée. Ce processus a été présenté brièvement dans la section 2.3.. Les vecteurs de représentation ont temporairement une dimension supérieure à la dimension finale souhaitée mais retrouvent la bonne dimension à l'étape d'itération suivante.
Nous avons ici utilisé deux *apriori* sur la matrice document-mots. Le premier est la matrice réduite document-mots-clefs. Ajoutée à la matrice de représentation des documents, elle permet de favoriser la convergence des documents partageant les mêmes mots-clefs. Le second *apriori* est la matrice de cooccurrences réduite issue de la base wikipédia en français. Elle favorise le rapprochement de mots cooccurrant avec les mêmes mots voisins, bien que cette information ne soit pas disponible dans la matrice document-mots principale.
Comme le montre la figure 4.5, l'apport de ces *apriori* est mitigé. Ils améliorent les performances pour les corpus de linguistique et de sciences de l'information, mais n'ont pas d'effet sur les deux autres corpus, voire des effets négatifs pour des seuils de sélection finaux supérieurs à l'optimal.

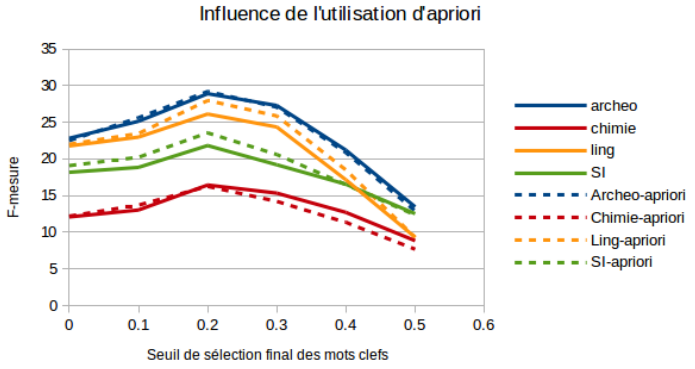

FIGURE 4.5. *Influence du l'utilisation d'apriori selon le seuil de sélection final des mots-clefs.*

4.6. Résultats

Cette approche sémantique est donc au final stable vis-à-vis des paramètres du kNN. Le paramètre important est le seuil de sélection final des mots-clefs qui donne les meilleurs résultats en terme de F-mesure à 0,2 pour les quatre corpus. L'ajout d'*apriori* lors de la factorisation de la matrice document-mots avant l'emploi du kNN a un effet faible, légèrement positif. Le tableau 4.8 présente les résultats obtenus avec ces paramètres [5].

Si le but est d'optimiser la précision, cette approche graphique obtient également de bons résultats, comme le présente le tableau 4.9 (paramètre de sélection à 0,4).

5. Les résultats présentés dans les actes de DEFT sont en fait optimisés pour la précision dans le tableau équivalent.

	Précision	Rappel	F-mesure
Archéologie	38,4	27,1	29,2
Linguistique	33,5	27,9	28,0
Sciences de l'information	27,5	26,1	23,6
Chimie	19,8	17,6	16,3

Tableau 4.8. *Scores obtenus par similarité sémantique entre documents voisins en optimisant la F-mesure.*

	Précision	Rappel	F-mesure
Archéologie	63,6	13,4	20,7
Linguistique	45,6	12,8	18,5
Sciences de l'information	40,7	11,6	15,9
Chimie	31,1	7,9	11,1

Tableau 4.9. *Scores obtenus par similarité sémantique entre documents voisins en optimisant la précision.*

4.7. Discussion

En terme de F-mesure, les résultats obtenus par l'approche sémantique sont meilleurs que ceux obtenus par l'approche graphique pour les corpus de linguistique et de sciences de l'information. Cette augmentation est essentiellement dûe à une amélioration drastique de la précision. À l'inverse, l'approche graphique est meilleure sur les corpus de chimie et surtout d'archéologie.

Pour le corpus d'archéologie, la meilleure performance de la méthode graphique s'explique par le fait que 63 % des mots-clefs sont présents dans les notices (voir tableau 4.10). Pour le corpus de chimie, la faible proportion de mots-clefs du corpus de test déjà présents dans le corpus d'entraînement pénalise la méthode sémantique. Aussi, la combinaison des deux approches devrait apporter une certaine stabilité aux résultats quel que soit le schéma d'annotation privilégié dans un corpus.

	Proportion des mots-clefs des documents tests déjà présents dans les documents d'entraînement	Proportion de mots-clefs présents dans le document
Archéologie	60 %	63 %
Linguistique	58 %	39 %
Sciences de l'information	55 %	32 %
Chimie	44 %	24 %

Tableau 4.10. *Statistiques des mots-clefs.*

Parmi les propositions des autres équipes participantes à DEFT 2016, trois optent également pour une approche par assignation. (Bougouin *et al.*, 2016) utilise des graphes et des regroupements en sujets. Il s'agit de la seule méthode d'assignation obtenant une aussi forte précision sur le corpus d'archéologie. À l'inverse, notre méthode, ainsi que celles de (Chartier *et al.*, 2016b), utilisant des espaces sémantiques construits sur des graphèmes, et de (Buscaldi & Zargayouna, 2016), utilisant des vecteurs de cooccurrences pondérés par l'information mutuelle, sont comparativement plus précises pour les corpus de linguistique et de sciences de l'information.
Il est à noter que (Buscaldi & Zargayouna, 2016) observent une amélioration de tous leurs résultats lorsqu'ils appliquent une LSA sur leur matrice de cooccurrence, ce qui nous conforte dans notre choix d'utilisation de notre algorithme NC-ISC sur nos propres matrices.

5 Combinaison des approches précédentes

Chacune des approches précédentes fournit une liste de mots-clefs candidats associés à un score de similarité. Certains mots-clefs se retrouvent dans les deux listes mais ces dernières sont rarement exactement semblables,

comme on peut le voir au tableau 5.11. Notamment, par construction, les mots-clefs issus du modèle graphique sont limités à ceux présents dans la notice. Quant à ceux issus du modèle sémantique, ils sont limités aux mots-clefs déjà utilisés dans le corpus annoté d'entraînement. Il est donc intéressant de chercher à mixer les deux listes.

	spécifiques graphique	spécifiques sémantique	communs
Archéologie	57,0 %	20,5 %	22,5 %
Linguistique	30,4 %	48,0 %	21,6 %
Sciences de l'information	38,4 %	39,5 %	22,1 %
Chimie	44,2 %	41,8 %	14,0 %

Tableau 5.11. *Chevauchement des mots-clefs correctement identifiés, moyenne des pourcentages.*

Cependant, les scores associés aux mots-clefs, issus de deux modèles différents, calculés sur deux matrices différentes, ne sont pas directement comparables.

Une approche simple pour les rendre compatibles est de les mettre à l'échelle : les scores de similarité des mots-clefs issus du modèle graphique, plus élevés, sont multipliés par le score de similarité du meilleur mot-clef issu du modèle sémantique. Les résultats sont ensuite fusionnés. Si un mot-clef est proposé à la fois par le modèle sémantique et le modèle graphique, alors leurs scores sont additionnés. Nous avons également testé l'emploi d'une régression logistique avec un seuil de décision sur les scores calculés.

5.1. Etalonnage des scores

5.1..1. Mise à l'échelle directe

Dans l'approche graphique, le paramètre le plus important est le seuil de similarité de sélection, appelé int et fixé empiriquement à 0,98. Pour l'approche sémantique, le paramètre influençant le plus les résultats est le seuil de sélection final, appelé res et fixé à 0,2. Ces paramètres sont optimaux lorsque les méthodes fonctionnent seules. Nous avons étudié leur effet combiné. La figure 5.6 présente l'évolution des performances en fonction de la variation de res avec int fixé à 0,98. La figure 5.7 quant à elle présente la variation de int avec res fixé à 0,2.

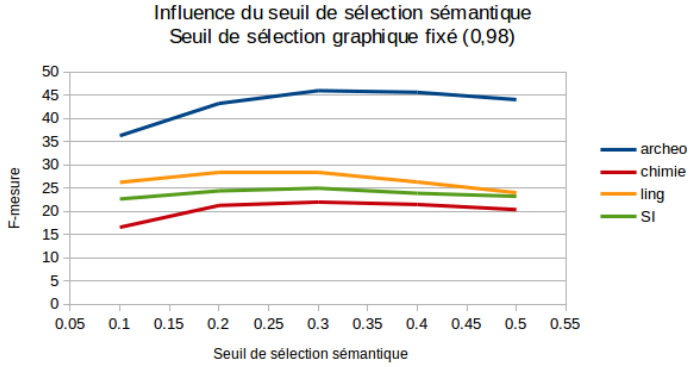

FIGURE *5.6.* *Evolution en fonction de res pour int = 0,98.*

Plus int augmente et plus les résultats augmentent. Cependant, il n'y a pas de différence significative entre 0,98 et 0,99. L'influence de res est plus marquée. Alors que sa valeur optimale était de 0,2 lorsque la méthode sémantique était employée seule, elle est de 0,3 employée en combinaison avec la méthode graphique. Comme les deux méthodes sélectionnent des mots-clefs différents, leur combinaison va augmenter le rappel, au risque de diminuer la précision. Il est donc compréhensible que les paramètres optimaux dans le cadre du mixage soient décalés par rapport à l'utilisation solitaire dans le sens de plus de précision. Chaque liste individuelle aura un moins bon rappel, largement compensé par leur combinaison.

Les résultats présentés au tableau 5.12 montrent que le mixage des deux méthodes dans sa version simple atteint son but. Le rappel obtenu est bien meilleur que pour chaque méthode utilisée isolément, tout en gardant une précision correct. Cela mène donc à une amélioration en terme de F-mesure.

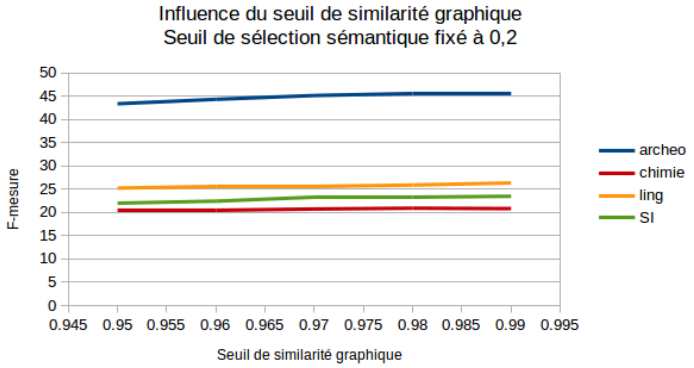

FIGURE 5.7. *Evolution en fonction de int pour res = 0,2.*

	Précision	Rappel	F-mesure
Archéologie	41,6	56,1	45,9
Linguistique	24,2	37,5	28,4
Sciences de l'information	21,1	34,3	24,9
Chimie	22,9	24,5	21,9

Tableau 5.12. *Scores obtenus en combinant les deux approches.*

5.1..2. Combinaison par régression logistique

Une autre façon de mixer les résultats est simplement d'appliquer une régression logistique. Afin de développer un outil le plus robuste possible vis à vis des changements de sujet, nous avons entraîné un seul classifieur sur les données des quatre corpus. A partir des mots-clefs sélectionnés sur le corpus d'entraînement par les méthodes graphiques et sémantiques et des scores associés, nous avons réalisé une régression logistique avec l'implémentation python *LogisticRegressionCV* de *sklearn.linear_model*. Le classifieur ainsi constitué peut décider si un mot-clef proposé doit être maintenu à partir des scores que lui attribuent les méthodes graphique et sémantique. Le classifieur est ensuite appliqué aux mots-clefs proposés par les méthodes sur les documents test. Les résultats sont rassemblés dans le tableau 5.13.

	Précision	Rappel	F-mesure
Archéologie	43,3	57,9	47,7
Linguistique	25,2	38,4	29,2
Sciences de l'information	22,1	32,6	25,1
Chimie	23,3	25,3	22,6

Tableau 5.13. *Scores obtenus en combinant les deux approches par régression logistique.*

5.2. Comparaison avec la LSA

Notre méthode de factorisation de matrice se rapproche beaucoup de la LSA. Nous avons donc comparé les performances obtenues avec des matrices réduites avec cette méthode. La figure 5.8 montre que si NC-ISC obtient de meilleurs résultats sur le corpus archéologie, il n'y a pas de différence significative pour les trois autres corpus. Toute la méthodologie que nous exposons dans cet article peut donc être appliquée avec d'autres algorithmes de création d'espaces vectoriels denses. Cependant, NC-ISC à l'avantage d'être performant avec peu de traits finaux et d'être facilement distribuable et rapide d'exécution, ce qui est un avantage pour traiter des corpus volumineux.

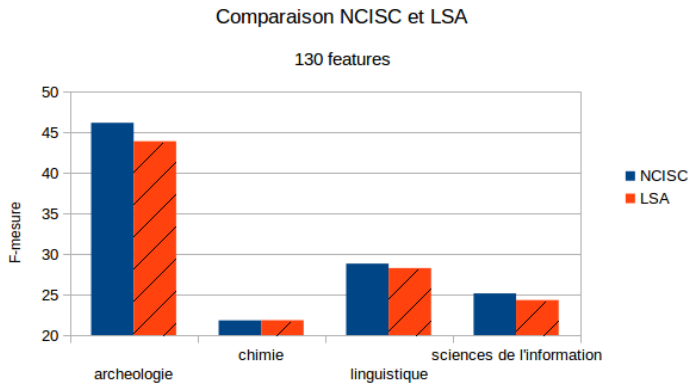

FIGURE 5.8. *Comparaison entre NC-ISC et LSA à nombre de dimensions fixé.*

6 Discussion des résultats

Le tableau 6.14 rassemble pour comparaison les trois méthodes, chacune optimisée pour la F-mesure. On remarque que pour les corpus de chimie et d'archéologie, la méthode graphique est meilleure que la méthode sémantique, alors que pour ceux de linguistique et de sciences de l'information, c'est la méthode sémantique qui l'emporte.

Comme indiqué au tableau 4.10, plus la proportion de mots-clefs présents dans le corpus du document est élevée, plus la performance de la méthode graphique l'est aussi. Cet état de fait est particulièrement marqué pour le corpus d'archéologie qui présente dans le corps des textes 63 % des mots-clefs choisis. La même relation s'observe entre la proportion des mots-clefs des documents de test déjà présents dans les documents d'entraînement et la performance de la méthode sémantique.

Pour les quatre corpus, la méthode sémantique, optimisée pour la F-mesure, donne une meilleure précision que rappel. Nous avons d'ailleurs vu dans la partie qui lui est consacrée que cette méthode peut obtenir des scores de précision élevés si on l'optimise pour cela. Utiliser des représentations vectorielles denses de documents associées à un algorithme de plus proches voisins est en effet très performant pour des tâches de classification. Ici, comme il s'agit d'indexation, la méthode est pénalisée par le grand nombre d'étiquettes et surtout d'étiquettes inconnues.
Nos tests comparatifs entre NC-ISC et LSA montrent un léger avantage, rarement significatif, pour NC-ISC à nombre de dimensions fixé. NC-ISC a cependant l'avantage d'être extrêmement rapide d'exécution. Il pourrait être intéressant de poursuivre la comparaison avec d'autres méthodes de représentation vectorielle de documents comme Glove (Pennington *et al.*, 2014) ou doc2vec (Le & Mikolov, 2014). Cependant, le prétraitement du corpus a souvent plus d'influence sur les performances que la méthode utilisée. Il serait donc intéressant d'observer la performance de cette méthode en utilisant des vecteurs calculés sur des textes en sac de lemmes ou même sac de graphèmes.

À l'inverse, la méthode graphique donne en général un meilleur rappel que précision. La seule exception est le corpus chimie qui présente une précision élevée. Comme nous l'avons vu à la partie 3, c'est bien l'utilisation de graphèmes qui permet ce fort rappel, ce qui n'est pas le cas si on utilise simplement des lemmes. La proximité des vecteurs graphiques permet de rapprocher des mots qui sont de la même famille sans pour autant partager un même lemme ou racine. Cela peut cependant amener une précision plus faible. Si la précision est plus élevée pour le corpus chimie, c'est sans doute parce que ce corpus emploie comme mots-clefs des noms de molécules. Or, même si certains noms de molécules partagent des sous-chaînes en commun, les sous-chaînes spécifiques sont suffisamment longues pour empêcher les vecteurs les représentant d'être suffisamment proches pour être assimilés.
Telle qu'implémentée, la méthode graphique cherche une correspondance dans les textes de tous les mots pleins des mots-clefs. Or, certains mots génériques d'un domaine peuvent très bien être employés dans un mot-clef sans apparaître dans le corps du texte. Par exemple, un texte de chimie ayant pour mot-clef "molécule alcoolique" pourra ne pas utiliser le mot "molécule" dans sont résumé mais seulement parler "d'alcools". Il pourrait être intéressant de repérer les mots peu informatifs dans les mots-clefs comportant plusieurs

mots, et de ne pas en tenir compte lors de la recherche des mots pleins dans le texte. Ces mots peu informatifs peuvent être différents selon le domaine. Ils pourraient être repérés par des mesures de tf-idf ou de spécificité.

La combinaison des deux méthodes, par mise à l'échelle simple des scores ou régression logistique, présente une précision intermédiaire entre les deux méthodes graphique et sémantique et un rappel supérieur aux deux. Nous avons en effet vu au tableau 5.11 que les listes de mots-clefs proposés par les deux méthodes sont en grande partie disjointes. Ce comportement a également été observé par l'équipe (Buscaldi & Zargayouna, 2016) lorsqu'ils ont combiné leur approche extractive à leur approche attributive.
Au final, la méthode mixte utilisant la régression logistique présente des scores de F-mesure supérieurs aux deux méthodes simples pour les quatre corpus. Utiliser la méthode mixte permet donc d'être sûr d'obtenir des résultats convenables en terme de F-mesure, quelle que soit l'habitude d'indexation dans le corpus de spécialité considéré. Il pourrait être intéressant de poursuivre les travaux en étudiant d'autres techniques de mixage d'experts.

		Précision	Rappel	F-mesure
Archéologie	graphique	**43,5**	50,0	44,4
	sémantique	38,4	27,1	29,2
	mixte simple	41,6	56,1	45,9
	mixte logit	**43,3**	**57,9**	**47,7**
Linguistique	graphique	20,6	26,0	22,0
	sémantique	**33,5**	27,9	28,0
	mixte simple	24,2	37,5	28,4
	mixte logit	25,2	**38,4**	**29,2**
Sciences de l'information	graphique	19,7	23,6	20,4
	sémantique	**27,5**	26,1	23,6
	mixte simple	21,1	**34,3**	**24,9**
	mixte logit	22,1	32,6	**25,1**
Chimie	graphique	**25,5**	17,8	19,3
	sémantique	19,8	17,6	16,3
	mixte simple	22,9	24,5	**21,9**
	mixte logit	23,3	**25,3**	**22,6**

Tableau 6.14. *Comparaison des trois approches, chacune étant optimisée pour la F-mesure.*

Lors de ce défi, sur cinq participants, notre méthode s'est classée première sur les corpus de chimie et d'archéologie, deuxième sur ceux de sciences de l'information et de linguistique, validant notre approche statistique. La figure 6.9 présente les résultats des différentes équipes sur les quatre corpus. Notre contribution est présentée en bleu foncé. En bleu rayé sont présentés nos scores actuels. La correction de la normalisation des textes, amenant à une meilleure performance du modèle graphique, et l'utilisation d'une régression logistique pour le couplage, ont permis d'améliorer le rappel, et par conséquent la F-mesure. Ces modifications ne changent pas le classement des différentes méthodes. On constate que, comparativement aux méthodes des autres participants, notre point fort est surtout le rappel. Notre statégie de combinaison d'indexations extractive et attributive est donc bien performante de ce point de vue.

7 Conclusion

Cet article présente la contribution d'eXenSa à DEFT 2016 sur la tâche d'indexation de notices bibliographiques par des mots-clefs.
Notre approche statistique combine deux parties, l'une graphique et l'autre sémantique. La première cherche dans la notice du document des mots graphiquement proches des mots-clefs usités des thésaurus de spécialité. La seconde attribue à un nouveau document les mots-clefs associés aux documents du corpus d'entraînement qui lui sont sémantiquement les plus proches. Les deux approches utilisent des représentations vectorielles apprises en utilisant notre algorithme NC-ISC, un algorithme stochastique de factorisation de matrices.
À des fins de généralisation, nous utilisons le même corpus d'entraînement et le même paramétrage pour tous les

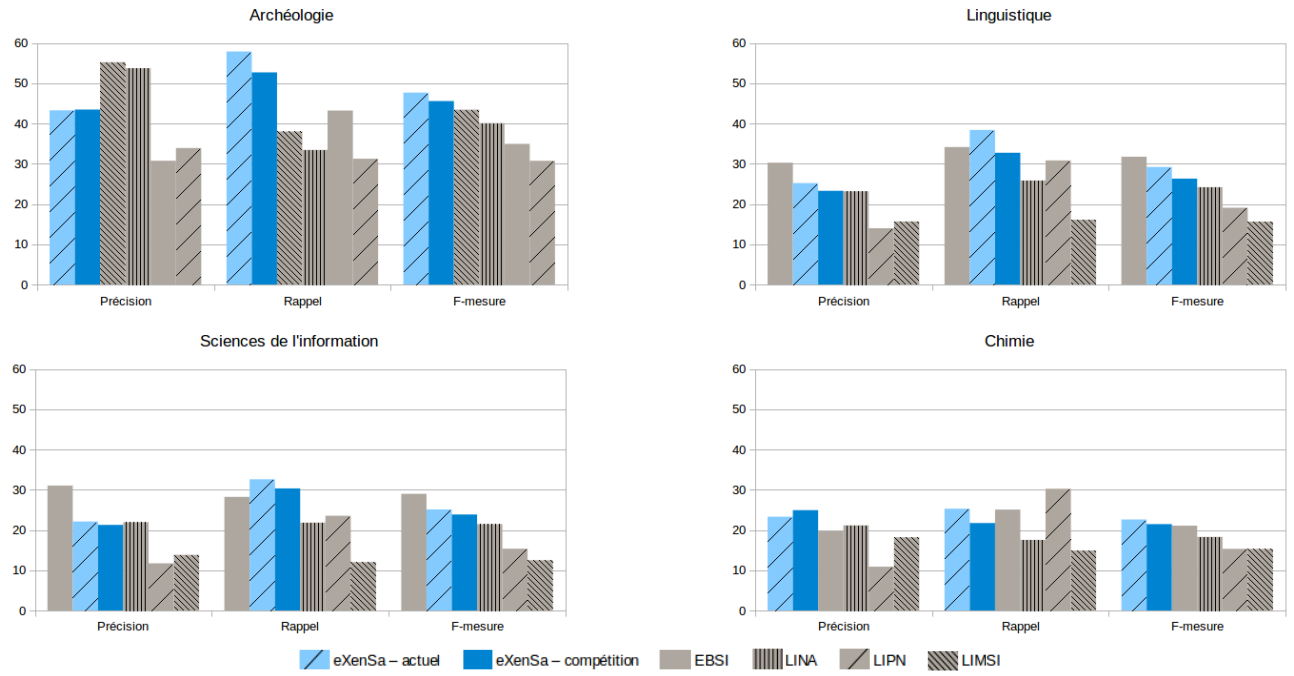

FIGURE *6.9.* *Résultats obtenus par les différentes équipes.*
Bleu foncé : notre résultat obtenu lors de la compétition. Bleu rayé : notre résultat actuel.
EBSI : (Chartier et al., 2016b) ; LINA : (Bougouin et al., 2016) ; LIPN : (Buscaldi & Zargayouna, 2016) ; LIMSI : (Hamon, 2016)

domaines de spécialité. Notre système final se classe premier sur deux des quatre corpus de test, deuxième sur les autres. De plus, une fois l'apprentissage effectué, l'attribution de mots-clefs à un nouveau document ne prend que quelques millisecondes. Une approche statistique est donc bien apropriée pour cette tâche bien qu'elle soit ici pénalisée par la petite taille des corpus.
Par construction, notre système n'attribue que des mots-clefs déjà présents dans les corpus d'entraînement. La proportion de mots-clefs du corpus d'apprentissage effectivement présents dans le corpus de test a donc un impact majeur sur les résultats. La proportion de mots-clefs présents dans les résumés a aussi bien évidemment une influence sur la partie graphique de la méthode. Le mixage des deux approches permet d'assurer une F-mesure convenable, quelle que soit l'habitude d'indexation d'un domaine de spécialité spécifique.

Références

AHAT M., PETERMANN C., HOAREAU Y. V. & BEN S. (2012). Algorithme automatique non supervisé pour le deft 2012. p.72.

BARKER K. & CORNACCHIA N. (2000). Using noun phrase heads to extract document keyphrases. *Advances in Artificial Intelligence*, p. 40–52.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv :1607.04606*.

BOUDIN F., HAZEM A., HERNANDEZ N. & SHRESTHA P. (2012). Participation du lina à deft 2012. p. 61–68.

BOUDIN F. & MORIN E. (2013). Keyphrase extraction for n-best reranking in multi-sentence compression. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

BOUGOUIN A., BOUDIN F. & DAILLE B. (2016). Topicrank en domaines de spécialité : participation du lina à deft2016.

BUSCALDI D. & ZARGAYOUNA H. (2016). Lipn@ deft2016 : Annotation de documents en utilisant l'information mutuelle. In *DÉfi Fouille de Texte 2016–DEFT2016*.

CHARTIER J.-F., FOREST D. & LACOMBE O. (2016a). Alignement de deux espaces sémantiques à des fins d'indexation automatique. *PARIS Inalco du 4 au 8 juillet 2016*, p.1̃3.

CHARTIER J.-F., FOREST D. & LACOMBE O. (2016b). Alignement de deux espaces sémantiques à des fins d'indexation automatique.

CLAVEAU V. (2012). Vectorisation, okapi et calcul de similarité pour le tal : pour oublier enfin le tf-idf. In *TALN-Traitement Automatique des Langues Naturelles*.

CLAVEAU V. & RAYMOND C. (2012). Participation de l'irisa à deft2012 : recherche d'information et apprentissage pour la génération de mots-clés. p. 49–60.

DAILLE B., BARREAUX S., BOUDIN F., BOUGOUIN A., CRAM D. & HAZEM A. (2016). Indexation d'articles scientifiques présentation et résultats du défi fouille de textes deft 2016. *Actes de 12e Défi Fouille de Texte (DEFT)*, p. 1–12.

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(6), 391.

DING Z., ZHANG Q. & HUANG X. (2011). Keyphrase extraction from online news using binary integer programming. In *IJCNLP*, p. 165–173.

DOUALAN G., BOUCHER M., BRIXTEL R., LEJEUNE G. & DIAS G. (2012). Détection de mots-clés par approches au grain caractère et au grain mot. p. 45–52.

DRINEAS P. & MAHONEY M. W. (2016). Randnla : randomized numerical linear algebra. *Communications of the ACM*, 59(6), 80–90.

D'AVANZO E. & MAGNINI B. (2005). A keyphrase-based approach to summarization : the lake system at duc-2005. In *Proceedings of DUC*.

EL GHALI A., HROMADA D. & EL GHALI K. (2012). Enrichir et raisonner sur des espaces sémantiques pour l'attribution de mots-clés. volume 2012, p.7̃7.

HALKO N., MARTINSSON P.-G. & TROPP J. A. (2011). Finding structure with randomness : Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2), 217–288.

HAMON T. (2012). Acquisition terminologique pour identifier les mots clés d'articles scientifiques. p.2̃8.

HAMON T. (2016). Indexation automatique de notices bibliographiques à l'aide d'approches d'acquisition terminologique.

HAN J., KIM T. & CHOI J. (2007). Web document clustering by using automatic keyphrase extraction. In *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops*, p. 56–59 : IEEE Computer Society.

HASAN K. S. & NG V. (2014). Automatic keyphrase extraction : A survey of the state of the art. In *ACL (1)*, p. 1262–1273.

ISHII N., MURAI T., YAMADA T. & BAO Y. (2006a). Text classification by combining grouping, lsa and knn. In *Computer and Information Science, 2006 and 2006 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse. ICIS-COMSAR 2006. 5th IEEE/ACIS International Conference on*, p. 148–154 : IEEE.

ISHII N., MURAI T., YAMADA T., BAO Y. & SUZUKI S. (2006b). Text classification : combining grouping, lsa and knn vs support vector machine. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, p. 393–400 : Springer.

JACQUEMIN C. (1997). Variation terminologique : Reconnaissance et acquisition automatiques de termes et de leurs variantes en corpus. *Mémoire d'Habilitation à Diriger des Recherches en informatique fondamentale, Université de Nantes*.

JONES S. & STAVELEY M. S. (1999). Phrasier : a system for interactive document retrieval using keyphrases. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, p. 160–167 : ACM.

KIM S. N., MEDELYAN O., KAN M.-Y. & BALDWIN T. (2010). Semeval-2010 task 5 : Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, p. 21–26 : Association for Computational Linguistics.

LANCASTER F. W., LANCASTER F. W., LANCASTER F. W. & LANCASTER F. W. (1991). *Indexing and abstracting in theory and practice*. Library Association London.

LANDAUER T. K. & DUMAIS S. T. (1997). A solution to plato's problem : The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2), 211.

LE Q. V. & MIKOLOV T. (2014). Distributed representations of sentences and documents. In *ICML*, volume 14, p. 1188–1196.

LEVY O. & GOLDBERG Y. (2014a). Dependency-based word embeddings. In *ACL (2)*, p. 302–308.

LEVY O. & GOLDBERG Y. (2014b). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, p. 2177–2185.

LEVY O., GOLDBERG Y. & DAGAN I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3, 211–225.

LITVAK M. & LAST M. (2008). Graph-based keyword extraction for single-document summarization. In *Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization*, p. 17–24 : Association for Computational Linguistics.

LIU Z., LI P., ZHENG Y. & SUN M. (2009). Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 1-Volume 1*, p. 257–266 : Association for Computational Linguistics.

MANNING C. D., SCHÜTZE H. *et al.* (1999). *Foundations of statistical natural language processing*, volume 999. MIT Press.

MATSUO Y. & ISHIZUKA M. (2004). Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(01), 157–169.

MEDELYAN O. & WITTEN I. H. (2006). Thesaurus based automatic keyphrase indexing. In *Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*, p. 296–297 : ACM.

MEDELYAN O. & WITTEN I. H. (2008). Domain-independent automatic keyphrase indexing with small training sets. *Journal of the Association for Information Science and Technology*, 59(7), 1026–1040.

MIHALCEA R. & TARAU P. (2004). : Association for Computational Linguistics.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

MIKOLOV T., YIH W.-T. & ZWEIG G. (2013c). Linguistic regularities in continuous space word representations. In *Hlt-naacl*, volume 13, p. 746–751.

MOREAU F., CLAVEAU V. & SÉBILLOT P. (2007). Automatic morphological query expansion using analogy-based machine learning. In *European Conference on Information Retrieval*, p. 222–233 : Springer.

PAROUBEK P., ZWEIGENBAUM P., FOREST D. & GROUIN C. (2012). Indexation libre et contrôlée d'articles scientifiques présentation et résultats du défi fouille de textes deft2012.

PARTALAS I., GAUSSIER É. & NGOMO A.-C. N. (2013). Results of the first bioasq workshop. In *BioASQ@ CLEF*, p. 1–8.

PENNINGTON J., SOCHER R. & MANNING C. (2014). Glove : Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, p. 1532–1543.

SALTON G., WONG A. & YANG C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11), 613–620.

SAVOY J. (2005). Indexation manuelle et automatique : une évaluation comparative basée sur un corpus en langue française. In *Actes de la 2ème Conférence en Recherche d'Information et Applications CORIA'05*, p. 9–23 : Institut d'Informatique et Mathématique Appliquée de Genoble, Laboratoire CLIPS (Communication Langagière et Interaction Personne-Système).

SPARCK JONES K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1), 11–21.

TONELLI S., CABRIO E. & PIANTA E. (2012). Key-concept extraction from french articles with kx. p. 19–28.

WAN X. & XIAO J. (2008). Single document keyphrase extraction using neighborhood knowledge. In *AAAI*, volume 8, p. 855–860.

WAN X., YANG J. & XIAO J. (2007). Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In *ACL*, volume 7, p. 552–559.

WITTEN I. H., PAYNTER G. W., FRANK E., GUTWIN C. & NEVILL-MANNING C. G. (1999). Kea : Practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries*, p. 254–255 : ACM.

ZHANG M.-L. & ZHOU Z.-H. (2014). A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8), 1819–1837.

