Extraction automatique de termes-clés : Comparaison de méthodes non supervisées

Josiane Mothe, Faneva Ramiandrisoa

Extraction automatique de termes-clés :
Comparaison de méthodes non supervisées

Sous la direction de Josiane Mothe**

Faneva Ramiandrisoa*
* *LAMAI (Laboratoire de Mathématiques Appliquées et Informatique) Université d'Antananarivo, Madagascar. r.faneva.mahery@gmail.com*

** *IRIT (Institut de Recherche en Informatique de Toulouse), UMR5505 CNRS, Université de Toulouse, Toulouse, France.* josiane.mothe@irit.fr

RÉSUMÉ. *Cet article présente un état de l'art et une comparaison des méthodes non supervisées et automatiques d'extraction de mots-clés à partir des contenus textuels de documents. Nous évaluons plusieurs méthodes de la littérature sur deux corpus de documents en comparant des termes-clés extraits et ceux associés initialement aux documents. Nous avons pu constater que la méthode basée sur une mesure TF-IDF était celle qui renvoie les résultats les plus proches des mots-clés des auteurs des documents.*

ABSTRACT. *This article presents a state of the art and a comparison of unsupervised methods for automatic keywords extraction from documents. We evaluate several methods from the literature on two sets of documents by comparing the keywords extracted and those initially associated with documents. We found that the best method (the one that retrieves keywords the closest to the authors' keywords) is based on TF-IDF.*

MOTS-CLÉS : *extraction automatique de termes-clés, méthodes non supervisées, représentation*

KEYWORDS: *keywords extraction, unsupervised method, representation*

1. Introduction

Les outils et moteurs de recherche utilisent une représentation des documents pour permettre la recherche de l'information susceptible de répondre au besoin de l'utilisateur que celui-ci exprime au travers de sa requête. Cette représentation est construite pendant la phase d'indexation. Plusieurs formes d'indexation existent et parfois cohabitent dans les systèmes de classement et d'accès. L'*indexation manuelle analytique* vise à représenter des contenus à travers des termes-clés. Les documentalistes s'appuient sur un thésaurus et sur des règles d'indexation pour choisir les termes à associer à une ressource. Les auteurs des ressources sont également souvent appelés à associer des termes-clés aux documents qu'ils écrivent, en s'appuyant sur des hiérarchies de concepts ou en proposant des termes-clés libres. L'*indexation automatique* utilisée par les moteurs de recherche suit le processus suivant : extraction des mots du texte, suppression des mots-vides et outils de la langue, radicalisation pour limiter la variante des mots et pondération.

Quelle que soit la façon d'associer les termes-clés à un document, manuellement ou automatiquement, ils ont pour rôle de synthétiser au mieux le contenu des documents, soit pour aider à leur sélection à partir d'une requête qui contient ces termes-clés, soit pour aider l'utilisateur à choisir parmi un ensemble de documents ceux qui répondent le mieux à leur besoin. Depuis les années 2000, plusieurs travaux se sont intéressés à l'extraction automatique de termes-clés comme cela peut être fait manuellement. C'est à cette forme de termes-clés que s'intéresse ce présent article. Les méthodes de la littérature sont soit supervisées soit non supervisées.

Les méthodes supervisées, comme celle de Liu (Liu *et al*, 2011) et de Sarkar (Sarkar *et al,* 2010) ou encore celle de Witten (Witten *et al,* 1999), ont la capacité d'apprendre à partir d'exemples. Elles s'appuient sur une collection de documents déjà annotés en termes-clés dans la phase d'apprentissage. Cette contrainte nous a fait choisir les méthodes non supervisées. Dans les méthodes non supervisées, nous pouvons distinguer les méthodes s'appuyant sur une représentation sous forme de graphes comme TextRank (Mihalcea *et al,* 2004), SingleRank (Wan *et al,* 2008), TopicRank (Bougouin *et al*, 2013), Kcore et WKcore (Rousseau *et al,* 2015), de celles qui ne les utilisent pas comme les méthodes à base de regroupent ou de fréquence comme l'algorithme KeyCluster (Liu *et al,* 2009).

L'objectif de cet article est de comparer ces méthodes afin d'étudier leur complémentarité. Nous nous sommes centrés sur les méthodes non supervisés à base de graphe car elles sont plus facilement généralisables et ne nécessitent pas de phase d'apprentissage. Par ailleurs, un graphe présente de manière efficace et simple les mots d'un document et les relations qu'ils entretiennent. Nous utilisons la méthode TF-IDF comme référence. Dans la suite de cet article, la section 2 décrit les méthodes étudiées, la section 3 présente les résultats. La section 4 conclut cet article.

2. Etat de l'art

2.1. *Termes-candidats*

Les termes-candidats sont des unités textuelles pouvant devenir des termes-clés (groupes de mots). Dans nos expérimentations, c'est la méthode « les plus longues séquences de noms et d'adjectifs » qui a été utilisée pour l'extraction des termes candidats ; les méthodes citées ci-dessous sont appliquées à ces candidats.

2.2. *TF-IDF*

TF-IDF (ou Term Frequency – Inverse Document Frequency) (Sparck Jones, 1972) mesure le pouvoir discriminant d'un mot ou d'un groupe de mots dans un document donné. Essentiellement, cette technique mesure l'importance d'un certain terme dans un document par rapport aux autres documents de la même collection.

$TF - IDF(terme) = TF(terme) \times \log(\frac{N}{DF(terme)})$ avec : ***TF*** nombre d'occurrences du terme dans le document analysé ; ***DF*** nombre de documents

dans lequel le terme est présent et ***N*** nombre total de documents.

Cette mesure est utilisée pour pondérer les termes-candidats : plus la valeur TF-IDF d'un terme-candidat est élevée, plus celui-ci est important dans le document analysé. En prenant compte de tous les documents dans le corpus, cette méthode présente généralement de meilleurs résultats.

Les méthodes qui vont suivre sont des méthodes à base de graphe noté G (N, A) où N est l'ensemble des nœuds et A l'ensemble de ses arcs sortants et entrants. Chaque sommet du graphe représente un terme-candidat et la constitution des arêtes est propre à chaque méthode.

2.3. *TextRank*

TextRank est basée sur le calcul du score d'importance des sommets en utilisant le principe de vote ou de recommandation entre deux sommets (Mihalcea *et al,* 2004) et inspiré de l'algorithme Pagerank (Page *et al*, 1999). TextRank utilise une représentation efficace d'un document, elle peut aussi être utilisée pour faire des résumés automatiques d'un document. Malgré cela, elle possède un inconvénient : au lieu d'ordonner des termes-candidats, elle n'ordonne que des mots. Cette méthode repose sur les étapes suivantes : *construction du graphe, calcul des scores des sommets et extractions des termes clés.*

Construction du graphe : les termes-candidats initiaux sont des mots simples ; ils sont utilisés comme sommets du graphe. Deux sommets sont reliés par une arête s'ils co-occurrent dans une fenêtre de N mots.

Calcul des scores des sommets : Au départ, les scores de tous les sommets du graphe sont initialisés aléatoirement. Un algorithme de classement calcule les scores de chaque sommet à chaque itération et s'arrête lorsque le seuil donné est atteint. Le score d'un sommet C_i est calculé avec la formule $S(C_i) = (1-d) + d * \sum_{C_j \in In(C_i)} \frac{1}{|Out(C_j)|} S(C_j)$ avec : $In(C_i)$ l'ensemble des sommets qui pointent vers C_i ; $Out(C_j)$ l'ensemble des sommets que pointent C_j et d un facteur d'amortissement[1]. Dans le cas où le graphe G serait non orienté, $In(C_i) = Out(C_i)$.

La formule suivante qui est utilisée dans le cas où les arêtes reliant les sommets sont pondérées: $WS(C_i) = (1-d) + d * \sum_{C_j \in In(C_i)} \frac{w_{ji}}{\sum_{C_k \in Out(C_j)} w_{jk}} WS(C_j)$

Extraction des termes-clés : C'est à partir des sommets les plus importants (selon leur score par ordre décroissant) que sont choisis les mots-clés. Les séquences des mots (des sommets importants) adjacents dans le document constituent les termes-clés composés de plusieurs mots; les autres mots non adjacents dans les documents qui obtiennent les meilleurs scores sont également retenus comme mots-clés.

2.4. *SingleRank*

[1] Peut être défini entre 0 et 1

SingleRank est une modification de la méthode TextRank, la différence se trouve dans la pondération des arêtes du graphe de mots et dans l'extraction des mots-clés à partir des mots-candidats, c'est-à-dire le calcul des scores des termes (Wan et al, 2008). Dans la majorité des cas, cette méthode fournit de meilleurs résultats que TextRank. L'inconvénient de cette méthode est qu'elle favorise les termes-candidats les plus longs (c'est-à-dire les termes formés par de nombreux mots) tout en faisant monter les candidats redondants dans le classement. Cette méthode, comme TextRank, repose sur trois étapes : *construction du graphe, calcul des scores des sommets et extractions des termes clés.*

Construction du graphe : les terme-candidats peuvent être composés et chaque mot composant chaque terme-candidat est considéré comme sommet du graphe. Deux sommets sont reliés par une arête s'ils co-occurrent dans une fenêtre de N mots et le poids de cette arête est le nombre de cooccurrence de ces deux sommets.

Calcul des scores des sommets : SingleRank utilise un algorithme de classement pour calculer les scores des sommets avec la formule suivante : $S(C_i) = (1-\lambda) + \lambda \times \sum_{C_j \in A_{entrant}(C_i)} \frac{p_{j,i} \times S(C_j)}{\sum_{C_k \in A_{sortant}(C_j)} p_{j,k}}$ Avec : λ : facteur d'atténuation (peut être considéré comme la probabilité que le nœud C_i soit atteint par recommandation) ; $p_{j,i}$: poids de l'arc allant du nœud C_j vers le nœud C_i, correspondant au nombre de cooccurrences entre les deux mots i et j.

Les scores des termes-candidats constitués de plusieurs mots sont calculés par : $TermeScore(p_i) = \sum S(C_j)$ Avec : p_i est un terme-candidat constitué de plusieurs mots ; C_j est un des mots composant le terme-candidat p_i

Extraction des termes-clés : Ce sont les termes-candidats ayant les scores les plus importants qui sont retenus comme termes-clés. Il n'y a donc pas de génération de termes-clés comme cela est le cas dans TextRank.

2.5. *TopicRank*

TopicRank, fondée sur TextRank, est différente par rapport aux autres méthodes à base de graphe, parce qu'au lieu de faire une recherche des unités textuelles importantes du document, elle cherche ses sujets importants. Par rapport à TextRank et SingleRank, elle présente les avantages suivants : suppression des problèmes de redondance dans les termes-clés extraits, construction d'un graphe plus compacte, renforcement des poids des arêtes dans le graphe, amélioration de la qualité d'ordonnancement et suppression du paramètre de la fenêtre de co-occurrences. Cette méthode repose sur trois étapes : *identification des sujets, ordonnancement des sujets, sélection des mots-clés* (Bougouin *et al,* 2013).

Construction du graphe : ce sont les termes-candidats, composés de plusieurs mots ou non, qui représentent les sommets du graphe et tous les sommets sont reliés entre eux, nous avons un graphe complet.

Identification des sujets : un sujet est une information spécifique (le plus souvent) ou générale transportée au minimum par une unité textuelle. Deux termes-candidats C_1 et C_2 sont groupés à partir de la similarité de Jaccard : $sim(C_1, C_2) = \frac{\|C_1 \cap C_2\|}{\|C_1 \cup C_2\|}$

Avec : C_1 U C_2 : nombre de mots commun à C_1 et C_2; C_1 ∩ C_2 : nombre de mots composant C_1 et C_2.

Dès que la similarité entre toutes les paires de termes-candidats est connue, l'algorithme de classification ascendante hiérarchique est appliqué. Au début, chaque terme-candidat est considéré comme un groupe et puis les deux groupes présentant la plus forte similarité sont réunis en un seul. Ce regroupement est répété jusqu'à ce que le nombre (prédéfini) de groupes soit atteint. La similarité entre deux groupes est obtenue en calculant la similarité entre les termes-candidats composant chaque groupe. La valeur de similarité entre deux groupes peut être obtenue en choisissant parmi les trois méthodes suivantes : *(simple):* la plus grande valeur de similarité est retenue ; *(complète) :* la plus petite valeur de similarité est retenue ; *(moyenne) :* la moyenne de toutes les similarités est retenue

Une fois les sujets connus, un nouveau graphe est défini comme suit : G = (N, A) : N= ensemble des sujets du document et A= ensemble des liens entre les nœuds.

Ordonnancement des sujets : la pondération des arêtes est très importante durant cette étape. C'est la force du lien sémantique (Wan and Xiao, 2008) entre les nœuds du graphe (ou plutôt les sujets) qui est considérée comme poids d'une arête.

Pour représenter cette force sémantique, la distance entre les termes-candidats des sujets est utilisée. $poids(S_i, S_j) = \sum_{C_i \in S_i} \sum_{C_j \in S_j} dist\,(C_i, C_j)$ avec $dist(C_i, C_j) = \sum_{p_i \in pos(C_i)} \sum_{p_j \in pos(C_j)} \frac{1}{|p_i - p_j|}$ Avec : $poids(S_i,S_j)$ est le poids de l'arête entre les sujets S_i et S_j; $dist(C_i,C_j)$ est la force sémantique entre les termes-candidats C_i et C_j ; $pos(C_i)$ est la position du terme candidat C_i dans le document analysé et $pos(C_j)$ est la position du terme candidat C_j dans le document analysé.

L'ordonnancement est basé sur le principe de vote ou de recommandation, c'est à dire qu'un nœud (sujet) est très important s'il est fortement connecté avec plusieurs autres nœuds importants.

$importance(S_i) = (1 - \lambda) + \lambda \times \sum_{S_j \in V_i} \frac{poids(s_i,s_j) \times importance(s_j)}{\sum_{S_k \in V_j} poids(s_j,s_k)}$ Avec :

V_i est l'ensemble des sujets reliés au sujet S_i et λ est le facteur d'atténuation.

Sélection des termes-clés : chaque sujet important ne va fournir qu'un seul terme-clé. Pour le choix d'un terme-clé représentant le mieux un sujet, trois méthodes ont été proposées :

– *Première position* : le terme-candidat d'un sujet apparaissant le premier dans le document est sélectionné,

– *Fréquence* : le terme-candidat d'un sujet le plus fréquent dans le document analysé est sélectionné,

– *Centroïde :* le terme-candidat d'un sujet le plus similaire aux autres mot-candidat du même sujet est sélectionné.

2.6. *Kcore*

Kcore (Rousseau *et al,* 2015) est aussi une méthode d'extraction de termes-clés à base de graphe. La construction de son graphe de mots est semblable à celle de TextRank ou SingleRank. Contrairement aux méthodes à base de graphe que nous avons présentées ci-dessus, elle n'utilise pas le principe de vote ou de recommandation pour calculer les scores d'importance des sommets mais utilise l'algorithme de Batagelj et Zaveršnik (Batagelj *et al,* 2011). Le problème est qu'elle dépend de la fenêtre de co-occurrence de mots.

Construction du graphe : ce sont les termes-candidats, constitués de plusieurs mots ou non, qui représentent les sommets du graphe et deux sommets sont reliés si les termes-candidats représentant ces sommets co-occurrent dans une fenêtre de N mots. Ces liens peuvent être pondérés par le nombre de co-occurrences des mots qu'ils relient dans le document, on parle de WKcore , sinon (dans le cas d'un graphe non pondéré) ils seront tous pondérés par 1, on parle de Kcore .

Le degré d'un sommet C ∈ G dans G est noté $deg_G(C)$. Autrement dit, dans le cas où G est un graphe non orienté, $deg_G(C)$ est la somme des poids des arêtes adjacentes à C (poids unitaire dans le cas d'un graphe non pondéré). Dans le cas où G est orienté, la notion de degré se divise en deux : degré sortant et degré entrant qui correspondent respectivement au nombre de liens sortants et de liens entrants.
Un Kcore est un sous graphe $H_k = (v', \varepsilon')$ d'un graphe G = (N, A) où $v' \subseteq$ N et $\varepsilon' \subseteq$A tel que $\forall$ v ∈ N, $deg_{Hk}(v) \geq k$ et H_k est le sous-graphe maximal, c'est à dire qu'il ne peut pas être augmenté sans perdre cette propriété (Rousseau *et al,* 2015).

Une fois le graphe de mots construit, une décomposition Kcore avec l'algorithme de Batagelj et Zaveršnik (Batagelj *et al,* 2011) du graphe est réalisée. L'algorithme attribut un nombre n à chaque sommet du graphe Kcore auquel il appartient (si v ∈ Kcore alors n = k). Puis l'ordonnancement des sommets se fait par ordre décroissant des nombres n et les premiers sont retenus comme termes-clés.

3. Expérimentations et résultats

3.1. *Cadre expérimental*

Nous utilisons deux collections afin d'étudier et de comparer les méthodes présentées dans la section 2. Nous avons construit ces collections en interrogeant le Web of Sciences. La collection que nous appelons IPM correspond aux 689 articles publiés entre 1975-2015 dans le journal *Information Processing and Managment* ; il y a en moyenne 5,7 mots-clés par documents associés par les auteurs. La collection IRJ est constituée de 344 articles publiés entre 2000-2015 dans le journal *Information Retrieval Journal*; il a en moyenne 4,7 mots-clés.

Pour analyser les résultats des différentes méthodes, nous comparons les termes-clés extraits automatiquement et ceux associés manuellement par les auteurs. Nous mesurons l''efficacité des méthodes d'extraction par le Rappel[2] (R) lorsque l'on considère les 15 premiers termes. Cette mesure est définie par :

$R = \frac{Nombre_{match}}{Nombre_{ref}}$: avec : $Nombre_{match}$ nombre de termes-clés générés identiques à ceux formulés par les auteurs, $Nombre_{result}$ nombre total de termes-clés générés automatiquement et $Nombre_{ref}$ nombre total de termes-clés formulés par les auteurs.

Une comparaison entre les termes-clés fournis par les auteurs et les termes-clés extraits à partir des méthodes implémentées est réalisée. Dans les expérimentations, nous considérons les mots soit tels qu'ils apparaissent dans les documents, soit après racinisation[3] par l'algorithme de Martin Porter (Porter, 1980). Nous avons implémenté toutes ces méthodes et les avons comparées en s'inspirant du code de Bougouin (disponible sur GitHub[4]).

3.2. *Influence des autres paramètres des méthodes*

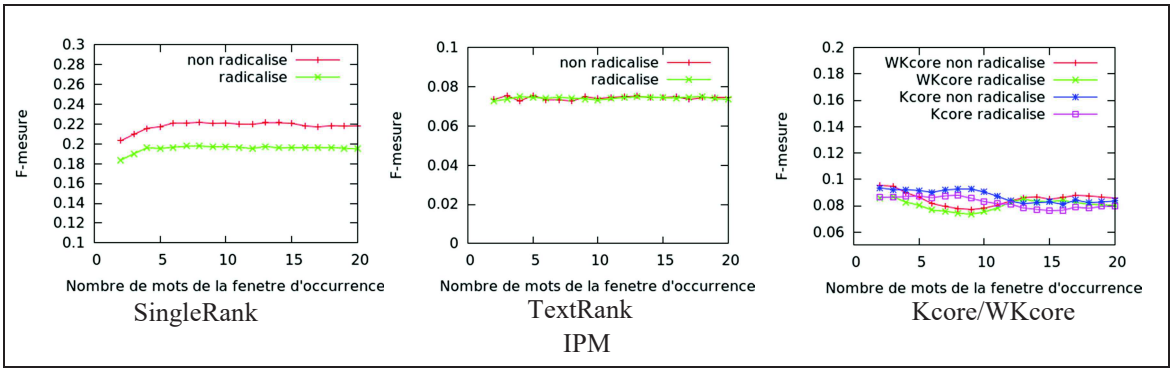

Figure 1. *Résultats de l'extraction avec SingleRank, TextRank, Kcore et WKcore en fonction de la fenêtre de cooccurrences utilisée, avec et sans racinisation.*

Le paramètre commun à TextRank, SingleRank, Kcore et WKcore est N, le nombre de mots dans la fenêtre de cooccurrences que nous faisons varier.

La figure 1 présente les résultats de SingleRank, TextRank, Kcore et WKcore sur le corpus IPM lorsque nous faisons varier la fenêtre de cooccurrence. Les meilleurs résultats sont obtenus par SingleRank. Pour SingleRank et TextRank, les résultats sont assez stables quelle que soit la valeur de N. La radicalisation diminue les résultats pour SingleRank alors qu'elle n'a que peu d'effet sur TextRank. En revanche, le paramètre N a plus d'influence sur les méthodes Kcore et WKcore. Kcore obtient de meilleurs résultats que WKcore pour N entre 6 et 11 alors que

[2] Rapport des résultats pertinents parmi les résultats retrouvés.
[3] Transformation d'un mot en sa racine c'est-à-dire suppression de ses suffixes et préfixes.
[4] https://github.com/adrien-bougouin/KeyBench/tree/ijcnlp_2013

WKcore devient meilleur lorsque N est plus important. Nous obtenons les mêmes résultats avec la collection IRJ.

Le paramètre de la méthode TopicRank est le seuil de similarité, la meilleure stratégie de groupement des termes-candidats et la meilleure stratégie de sélection du terme-candidat le plus représentatif d'un sujet. Nous avons fait varier le seuil de similarité ζ d'un pas de 0,10 pour toutes les stratégies de groupement (cf. figure 2).

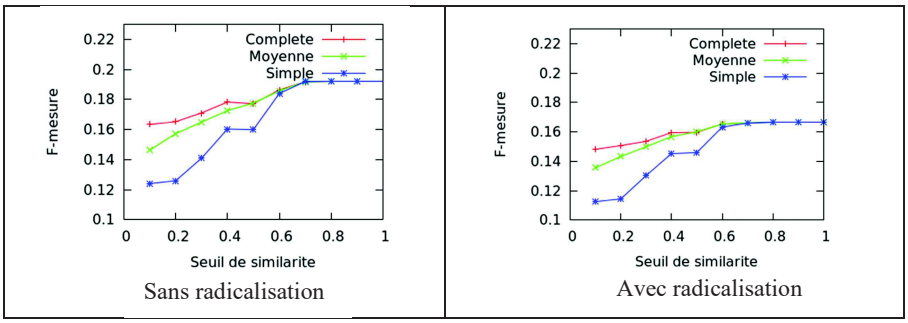

Figure 2. *Résultats de l'extraction de mots-clés avec TopicRank, en fonction de la stratégie de regroupement et de la valeur du seuil de similarité*

Les trois stratégies de regroupement ont chacune un comportement qui leur est propre jusqu'à un premier point de convergence dès que ζ > 0,50, ce point de convergence indique que plus d'un mot sur deux sont identiques dans les deux termes-candidats comparés. Le second point de convergence ζ = 0,70 correspond à strictement plus de deux mots sur trois sont identiques dans les deux termes-candidats comparés. Une fois passé ce deuxième point, les résultats sont stables ; cela correspond à la rareté des termes-clés composés de plus de 4 mots. Nous obtenons la même valeur de convergence pour la collection IRJ. Les meilleurs résultats sont obtenus en utilisant la stratégie complète, avec ou sans radicalisation.

Avec ou sans radicalisation, c'est le choix des candidats apparaissant en premier dans le document offre de meilleurs mots-clés que le choix des candidats centroïdes ou des candidats les plus fréquents ce qui confirme les résultats de Bouguoin (Bouguoin *et al,* 2013). La stratégie centroïde fournit de très faibles résultats par rapport aux deux autres stratégies.

3.3. *Comparaison des différentes méthodes*

Les valeurs des mesures dans le tableau suivant sont les moyennes des mesures obtenues sur l'ensemble les documents. Toutes les méthodes sont configurées de façon optimale c'est-à-dire dans l'optique d'obtenir les meilleurs termes-clés. Nous allons voir ainsi quelle est la méthode qui convient le mieux à notre cas et si la radicalisation des mots a une influence positive sur ces méthodes.

Méthodes	TF-IDF	TopicRank	TextRank	SingleRank	Kcore	WKcore

Rappel@5	0.13 (0.11)	0.13 (0.11)	0.03 (0.03)	0.11 (0.10)	0.03 (0.03)	0.04 (0.04)
Rappel@10	0.19 (0.17)	0.16 (0.14)	0.05 (0.05)	0.18 (0.16)	0.06 (0.06)	0.06 (0.06)
Rappel@15	0.22 (0.20)	0.19 (0.17)	0.08 (0.07)	0.22 (0.20)	0.09 (0.09)	0.10 (0.09)

Tableau 1. *Résultats de l'extraction des 5-10-15 mots-clés de plus fort poids selon les différentes méthodes, sans radicalisation et (avec radicalisation)*

TF-IDF fournit de meilleurs résultats que les autres méthodes, avec ou sans radicalisation. Il est aussi important à noter que c'est SingleRank qui fournit les meilleurs résultats parmi les méthodes basées sur un graphe de mots.

4. Conclusion

Nous avons constaté que globalement, TF-IDF fournit les mots-clés les plus proches de ceux formulés par les auteurs des articles ; ceci est possiblement dû au fait que cette méthode utilise tous les documents composant le corpus. Mais SingleRank, qui n'utilise que le document analysé, est l'une des meilleures méthodes d'extraction automatiques de mots-clés : ses performances ne sont pas éloignées de celles de TF-IDF. SingleRank serait d'autant plus performante qu'elle est utilisée sur un document complet au lieu d'un document composé seulement du titre et de son résumé (comme dans notre cas). L'utilisation de la radicalisation sur les documents avant l'extraction des mots-clés ne fournit pas de meilleurs résultats donc son intégration ne s'avère pas être utile dans notre analyse. Nous souhaitons maintenant confirmer sur de plus grand corpus de documents. Nous souhaitons également analyser la variation des valeurs des paramètres en fonction des collections et voir si les paramètres optimisés sur cette collection sont les mêmes sur d'autres collections.

La comparaison des performances en précision de ces méthodes d'extraction automatiques de mots-clés sont des comparaisons simples. Des tests statistiques seraient à utiliser dans le but de savoir si la différence de performance entre deux méthodes est significative ou non, surtout entre les méthodes TF-IDF et SingleRank.

Nous nous sommes intéressés à comparer les résultats de d'extraction automatique avec les mots choisis par les auteurs; ces méthodes pourraient également être utilisées pour suggérer des mots-clés aux auteurs.

5. Bibliographie

Batagelj V., Zaveršnik M., *Fast algorithms for determining (generalized) core groups in social networks*, Advances in Data Analysis and Classification, vol. 5, n° 2, 2011, p. 129-145.

Bougouin A., *Etat de l'art des méthodes d'extraction automatique de termes-clés*, In Rencontre des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RECITAL), 2013.

Bougouin A., Boudin F., and Daille B., *Topicrank: Graph-based topic ranking for keyphrase extraction*, In International Joint Conference on Natural Language Processing (IJCNLP), 2013. p. 543-551.

Brin S., Page L., *The anatomy of a large-scale hypertextual web search engine*, Computer networks and ISDN systems, vol. 30, n° 1, p. 107–117.

Liu Z., Chen X., Zheng Y., Sun M., *Automatic keyphrase extraction by bridging vocabulary gap*, In Proceedings of the Fifteenth Conference on Computational Natural Language Learning, Association for Computational Linguistics, 2011 p. 135-144.

Liu Z., Li P., Zheng Y., Sun M., *Clustering to find exemplar terms for keyphrase extraction*, In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Volume 1-Volume 1, 2009, p. 257-266.

Mihalcea R., Tarau P., *Textrank: Bringing order into texts*, Association for Computational Linguistics, 2004.

Page L., Brin S., Motwani R., Winograd T., *The PageRank citation ranking: bringing order to the Web, 1999*.

Porter M. F., *An algorithm for suffix stripping*, Program, vol. 14, n° 3, 1980, p. 130-137.

Rousseau F., Vazirgiannis M., *Main core retention on graph-of-words for single-document keyword extraction*, In Advances in Information Retrieval, 2015, p. 382–393.

Sarkar K., Nasipuri M., Ghose S., *A new approach to keyphrase extraction using neural networks*, arXiv preprint arXiv:1004.3274, 2010.

Sparck Jones K., *A statistical interpretation of term specificity and its application in retrieval*, Journal of documentation, vol. 28, n° 1, 1972, p. 11-21.

Wan X., Xiao J., *Single document keyphrase extraction using neighborhood knowledge*, In AAAI, vol. 8, 2008, p. 855–860.

Witten I. H., Paynter G. W., Frank E., Gutwin C., Nevill-Manning C. G., (1999, August). *KEA: Practical automatic keyphrase extraction*, In Proceedings of the fourth ACM conference on Digital libraries, ACM, p. 254-255.