Impact de la présence/absence des termes de la requête dans le document sur le processus d'appariement document-requête en utilisant Word2Vec

Thiziri BELKACEM[1] — Taoufiq DKAKI[2] — José G.MORENO[1], — Mohand BOUGHANEM[1]

[1] *Laboratoire IRIT, Université Paul Sabatier Toulouse 3*
[2] *Laboratoir IRIT, Université Jean Jaurès Toulouse 2*
{thiziri.belkacem, taoufiq.dkaki, jose.moreno, mohand.boughanem}@irit.fr

RÉSUMÉ. *Dans cet article, nous étudions l'appariement document-requête basé sur des similarités sémantiques entre les termes de la requête et ceux du document, à l'aide du plongement lexical des mots (word embedding). Contrairement aux approches traditionnelles qui sont basées sur les représentations dites sac de mots et qui reposent sur l'appariement exact entre les mots, le processus d'appariement pourrait être amélioré en tenant compte de tous les mots du document et en traitant différemment les mots de la requête qui ne sont pas dans le document. Nous avons exploité différentes stratégies d'appariement. Les résultats expérimentaux en utilisant des collections TREC montrent que les stratégies d'appariement étudiées donnent de meilleurs résultats que les modèles classiques de la RI.*

ABSTRACT. *In this paper we study a document-query matching based on semantic similarities between query and document terms using word embeddings. We show that unlike the traditional bag of words approaches, that rely on the exact matching between words, the matching process could be improved by taking into account all document terms and by processing differently query terms that are not in the document. We adopt different matching strategies that take into account the presence/absence of query terms in a document. Experimental results using TREC data sets show that the studied matching process outperforms the classical IR models.*

MOTS-CLÉS : *Similarités sémantiques, présence des termes de la requête, Plongement lexical.*

KEYWORDS: *Semantic similarities, Query terms presence, Word embeddings.*

1. Introduction

La limite principale des modèles sac de mots (BoW pour Bag of Words en Anglais) est liée à l'inadéquation du vocabulaire du document et celui de la requête, en particulier lorsque les termes de la requête ne sont pas utilisés dans les documents pertinents. Pour faire face à cette limite, des modèles basés sur des représentations vectorielles dites *plongement lexical* ou *word embedding* en Anglais, tels que LSI (Dumais *et al.*, 1988), PLSA (Hofmann, 1999) et plus récemment word2vec (Mikolov *et al.*, 2013a) et GloVe (Pennington *et al.*, 2014) ont été introduits. De tels modèles construisent des représentations contextuelles des mots capables de capturer leurs sens. Un mot est donc représenté comme vecteur de caractéristiques. Ces représentations, aussi dites représentations continues des mots, ont été exploitées en recherche d'information (RI) pour faire face à la limite de l'appariement exact. L'appariement document-requête est calculé en utilisant les similarités vectorielles des termes de la requête et du document, qui peuvent être considérées comme des similarités sémantiques. Construites à partir de similitudes lexicales pures de l'appariement des mots simples, ces représentations permettent une comparaison efficace entre les mots, puisque les distances entre deux vecteurs de mots sont sémantiquement significatives. La plupart des approches RI basées sur le plongement lexical des mots (Zamani et Croft, 2016 ; Kuzi *et al.*, 2016) font correspondre tous les termes de la requête avec tous les termes du document en utilisant indifféremment les similarités sémantiques ; ils ne font pas de distinction entre les termes de la requête qui sont présents dans le document et ceux qui ne le sont pas. Dans ce travail, notre objectif est d'étudier l'importance des termes de la requête qui ne sont pas présents dans les documents pertinents, dans le processus d'appariement document-requête basé sur les représentations continues des mots. Nous définissons les principales questions de recherche suivantes : Quel est l'influence des termes de la requête qui ne sont pas dans le document sur le processus d'appariement document-requête ? Devons-nous les traiter différemment ?

Pour répondre à ces questions, nous analysons différentes stratégies d'appariement document-requête. L'idée de base consiste à traiter les termes de la requête qui sont absents dans un document différemment de ceux qui y sont présents. Pour ce faire, nous combinons un modèle de RI classique avec des similarités sémantiques entre les termes de la requête et les termes du document. Nous avons étudié trois stratégies d'appariement différentes : 1) faire correspondre tous les vecteurs des termes de la requête, sans distinction, avec tous les vecteurs des termes du document. 2) l'appariement avec distinction des similarités des termes de la requête avec ceux d'un document en fonction de la présence ou non de ces termes dans ce document. 3) l'appariement qui, premièrement, traite différemment les termes de la requête qui apparaissent dans le document et ceux qui n'y sont pas présents, de plus, permet d'observer l'apport de l'appariement exact par rapport à l'apport de la similarité sémantique entre les mots du document et de la requête.

2. État de l'art

Les modèles de plongement lexical des mots, comme word2vec (Mikolov *et al.*, 2013b) et GloVe (Pennington *et al.*, 2014) construisent un espace de représentation sémantique distribuée des mots. Ces représentations sont largement utilisées en RI (Kenter et De Rijke, 2015 ; Kuzi *et al.*, 2016 ; Guo *et al.*, 2016) pour améliorer les modèles traditionnels. Kenter et al. (Kenter et De Rijke, 2015) combinent les caractéristiques sémantiques des mots issus de différents espaces de plongement lexical des mots pour l'appariement de textes courts. Le modèle entraîné est ensuite utilisé pour prédire la similarité sémantique pour les paires non étiquetées. Zamani et Croft (Zamani et Croft, 2016) proposent un modèle de langue qui construit le modèle de la requête en utilisant les représentations continues des mots. Le modèle utilise le plongement lexical des mots pour prédire les termes les plus appropriés pour l'expansion de la requête à partir des documents pertinents. Le modèle calcule par la suite le score d'appariement document-requête en utilisant la nouvelle requête ainsi construite et en supposant l'indépendance des similarité entre les termes vis-à-vis de la requête. Kuzi et al (Kuzi *et al.*, 2016) ont proposé différentes approches d'expansion des requêtes en utilisant les représentations continues des mots. Les documents sont ensuite classés en fonction du score de pertinence calculé par un modèle de langue classique par rapport à la nouvelle requête. Guo et al. (Guo *et al.*, 2016) ont proposé un modèle d'appariement sémantique dit *"modèle de transport non linéaire des mots"* (NWT pour *"Non-linear Word Transportation"*). Dans ce modèle, l'appariement entre le document et la requête est basé sur les similarités sémantiques entre les vecteurs des mots qu'ils contiennent. Les auteurs modélisent le processus d'appariement par un problème de transport non linéaire de tous les termes du document vers tous les termes de la requête. Le score de pertinence est ensuite calculé en fonction des flux de transport appris durant la phase d'entraînement.

Tous les modèles susmentionnés traitent de la même manière tous les termes de la requête : certains modèles comme (Zamani et Croft, 2016) utilisent tout le vocabulaire pour effectuer l'expansion de la requête en se basant sur les similarités sémantiques entre les vecteurs des mots. D'autres modèles (Guo *et al.*, 2016) représentent les documents et les requêtes sous forme d'un sac de vecteurs de mots (Bag of word embeddings) puis calculent le score de similarité en utilisant toutes les interactions entre les mots du document et ceux de la requête. Dans (Roy *et al.*, 2016), le modèle calcule le score de similarité document-requête en utilisant la distance moyenne entre les vecteurs des termes de la requête et le vecteur centroïde du document. Aucun des travaux précédents n'a analysé l'impact des termes de la requête qui sont absents dans un document et la façon dont ces termes contribuent au processus d'appariement. Dans cet article, nous montrons les résultats de l'étude menée sur l'impact de ces termes en utilisant les modèles de RI classique et les similarités sémantiques entre les vecteurs des mots.

3. Analyse de différentes stratégies pour l'appariement basé sur les liens sémantiques entre les mots

Dans les modèles BoW standards, le score de pertinence attribué au document D par rapport à la requête Q est généralement calculé en se basant sur l'appariement exact entre les termes q de la requête et les termes d du document et est calculée comme suit (Lv et Zhai, 2011) :

$$score(Q, D) = \sum_{q \in Q \cap D} u_{score}(q, D) \quad [1]$$

où $u_{score}(q, D)$ est le poids du terme q de la requête dans le document D, calculé en utilisant un modèle standard de RI.

Cette approche est basée sur l'appariement exact entre les termes de la requête et les termes du document, de sorte que les termes de la requête contribuent au score d'appariement document-requête uniquement s'ils sont présents dans le document, tandis que ceux qui n'y sont pas présents, même s'ils contribuent à la formulation de l'information recherchée par l'utilisateur avec la requête, ces termes ne sont pas considérés dans cette approche, cela peut entraîner l'omission de certaines informations ou la modification de l'information véhiculée par la requête.
Pour faire face à cette limite, les représentations distribuées des mots (Mikolov *et al.*, 2013b ; Pennington *et al.*, 2014) peuvent être utilisées. Ces représentations permettent d'exploiter les similarités sémantiques entre les vecteurs des différents mots, tels que les liens sémantiques entre les mots sont traduits par les distances entre les vecteurs correspondants.
Dans ce qui suit, nous allons décrire différentes stratégies permettant d'exploiter les liens sémantiques entre les vecteurs des mots de la requête et ceux du document, en se basant sur l'occurrence des mots de la requête dans les documents.

3.1. *Comparaison de tous les termes de la requête avec tous les termes du document en utilisant les similarités sémantiques*

Une façon simple est de comparer tous les termes de la requête avec tous les termes du document en utilisant leurs vecteurs représentatifs, comme dans (Guo *et al.*, 2016 ; Roy *et al.*, 2016). Ainsi, les poids des termes dans le document sont calculés en fonction de leurs similarités avec les termes de la requête. Dans ce cas, l'importance d'un terme dans le document considéré, est évalué par rapport aux termes de la requête. Ce processus est présenté dans l'équation 2 :

$$\begin{aligned} score(Q, D) &= \sum_{d \in D} \sum_{q \in Q} u_{score}(d, D) \times s(q, d)^{\alpha} \\ &= \sum_{d \in D} u_{score}(d, D) \times \sum_{q \in Q} s(q, d)^{\alpha} \end{aligned} \quad [2]$$

où $u_{score}(d, D)$ est le poids du terme d du document D dans celui-ci, $s(q, d)$ est la valeur normalisée de similarité sémantique entre le terme q de la requête Q et le terme d du document D. α est un paramètre utilisé pour contrôler l'impact de la similarité sémantique entre les mots.

En se basant sur la comparaison de tous les termes de la requête avec tous les termes du document dans l'équation 2, nous pouvons effectuer deux décompositions de manière à focaliser le processus d'appariement sur la notion de présence/absence des termes de la requête dans le document (équation 3) et/ou sur l'appariement exacte ou lexicale entre les termes (équation 4). Nous obtenons les deux stratégies d'appariement suivantes :

3.1.1. *Comparaison fractionnée en fonction de la présence/absence des terme de la requête dans le document*

Dans cette stratégie, nous avons l'intention d'observer la contribution des termes de la requête qui ne sont pas présents dans le document au calcul du score de pertinence de ce document. Nous décomposons l'équation 2 en deux parties : la première partie traite les termes de la requête qui sont présents dans le document D tandis que la deuxième partie concerne les termes de la requête qui ne sont pas dans le document. Ce processus est montré par l'équation 3 :

$$\begin{aligned} score(Q, D) = \lambda \times \sum_{d \in D} \sum_{q \in Q \cap D} u_{score}(d, D) \times s(q, d)^{\alpha} + \\ (1 - \lambda) \times \sum_{d \in D} \sum_{q \in Q \setminus D} u_{score}(d, D) \times s(q, d)^{\alpha} \end{aligned} \quad [3]$$

λ est utilisé pour contrôler l'impact de chacune des deux parties dans l'équation.

3.1.2. *Comparaison par séparation entre l'appariement exact u_{score} et l'appariement sémantique $s(.,.)$*

Nous avons décomposé la première partie de l'équation 3 en deux composantes, ce qui est montré dans l'équation 4. Dans cette équation, nous pouvons observer séparément l'impact des éléments suivants : l'appariement exact entre les termes de la requête et les termes du document, ce qui est décrit par la première partie de l'équation 4 ($\sum_{q \in Q \cap D} u_{score}(q, D)$). L'appariement sémantique entre les termes de la requête qui sont dans le document et les autres termes de celui-ci, ce qui est décrit par la deuxième partie de cette équation ($\sum_{q \in Q \cap D} \sum_{d \in D \setminus \{q\}} u_{score}(d, D) \times s(q, d)^{\alpha}$). La similarité sémantique entre les termes de la requête qui ne sont pas dans le

document avec les termes de celui-ci, ce qui est exprimé dans la troisième partie ($\sum_{d\in D}\sum_{q\in Q\setminus D} u_{score}(d,D)\times s(q,d)^{\alpha}$).

$$\begin{aligned} score(Q,D) = \lambda_1 \times \sum_{q\in Q\cap D} u_{score}(q,D) + \\ \lambda_2 \times \sum_{q\in Q\cap D}\sum_{d\in D\setminus\{q\}} u_{score}(d,D)\times s(q,d)^{\alpha} + \\ (1-\lambda_1-\lambda_2)\times \sum_{d\in D}\sum_{q\in Q\setminus D} u_{score}(d,D)\times s(q,d)^{\alpha} \end{aligned} \quad [4]$$

λ_1 et λ_2 sont des paramètres utilisés pour contrôler séparément les différentes interactions entre les termes de la requête et du document.

3.1.3. *Relation entre les différentes stratégies d'appariement*

Dans les stratégies d'appariement précédentes, notons que si $\lambda = 0.5$ dans l'équation 3 et si $\lambda_1 = \lambda_2 = 1/3$ dans l'équation 4 alors les équations 3 et 4 sont équivalentes à l'équation 2.

Soit la similarité entre deux mots w_i et w_j définis par l'équation 5 suivante :

$$s(w_i,w_j) = \begin{cases} 1 & si\ w_i = w_j \\ 0 & sinon \end{cases} \quad [5]$$

Si la similarité entre les termes de la requête et les termes du document est calculée par l'équation 5, dans chacune des stratégies d'appariement décrites par les équations 2, 3 et 4 alors l'équation 1 sera un cas particulier de ces équations : en utilisant l'équation 5 et si $\lambda = 0.5$ dans l'équation 3 et $\lambda_1 = \lambda_2 = 1/3$ dans l'équation 4 nous obtenons l'équation 1.

4. Expérimentations et résultats

Dans cette partie, nous allons d'abord décrire les modèles utilisés pour la pondération des termes des documents avec la fonction u_{score} dans les équations 2, 3 et 4. Ces modèles sont considérés comme des modèles de référence lorsqu'ils sont utilisés dans l'équation 1. Par la suite, nous décrirons le protocole expérimental en termes : des données de test dans la section 4.2, du réglage des paramètres et l'analyse de leur impact dans la section 4.3 puis des résultats obtenus et discussion en section 4.4.

4.1. *Les modèles de pondérations des termes*

4.1.1. *BM25*

C'est un modèle de pondération classique défini par Robertson et al (Robertson *et al.*, 1995). Pour nos tests, nous avons utilisé la définition suivante utilisé dans (Jones *et al.*, 2000) :

$$u_{score}(q,D) = \frac{(k_1 \times tf_{q,D})}{tf_{q,D} + k_1 \times \left(1 - b + \frac{b \times dl}{avgdl}\right)} \times idf_q \qquad [6]$$

$idf_q = log\frac{\|C\| - df_q}{df_q}$. Les paramètres dl et $\|C\|$ sont respectivement la taille du document en nombre de termes et la taille de la collection en nombre de documents. df_q est la fréquence du terme q de la requête, $avgdl$ est la longueur moyenne des documents et $tf_{q,D}$ est la fréquence du terme q dans le document D. k_1 et b sont fixés respectivement à 1.2 et 0.75 suivant le choix commun dans la littérature de la RI.

4.1.2. *Le modèle de langue (ML)*

Nous utilisons le modèle de langue défini dans (Metzler et Bruce, 2004), pour calculer la fonction u_{score} comme suit :

$$u_{score}(q,D) = log\left(\frac{\lambda_D \times \frac{tf_{q,D}}{dl}}{\lambda_C \times \frac{tf_{q,C}}{\|C\|}} + (1 - \lambda_D - \lambda_C) \times tf_{q,D} + 1\right) \qquad [7]$$

$tf_{q,C}$ est la fréquence du terme q de requête dans la collection C, $\lambda_D = 0.2$ et $\lambda_C = 0.4$ tels définis dans (Metzler et Bruce, 2004).

4.2. *Collection de données*

Nous avons utilisé les trois collections de données TREC décrites dans le tableau 1. La collection *AP 88-89* est composée des documents des disques 1 et 2 de la collections TREC, cette collection est utilisée pour définir les paramètres λ, λ_1, λ_2 et α pour chacune des stratégies d'appariement décrites dans la section 3. Les performances sont rapportées en utilisant les collections *Robust4* et GOV2. La première est composée de documents TREC du Disque 4 et du Disque 5 sans les documents du *Congressional Record*. La dernière collection est une grande collection web de 25 millions de documents provenant du domaine *gov*.

4.3. *Réglage des paramètres et analyse de leur impact*

Nous utilisons la fonction cosinus pour calculer la similarité $s(q,d)$ entre les termes du document et de la requête. Pour la représentation des mots, nous avons

Tableau 1 – Description de la collection utilisée

Collection de documents	Nombre de documents	Requêtes TREC correspondantes
AP 88-89	165 K	51 - 200
Robust4	528 k	301 - 450 601 - 700
GOV2	25 M	701 - 800

utilisé le modèle word2vec[1] pré-entraîné de Google dont chaque mot est représenté par un vecteur de 300 dimensions. Les termes qui ne font pas partie de ce modèle (dits OOV pour Out of Vocabulary terms) sont simplement ignorés dans le calcul de similarité document-requête. Soit $\overrightarrow{q}$ le vecteur du terme q de la requête et $\overrightarrow{d}$ le vecteur du terme d du document, la similarité entre q et d est calculée par l'équation 8 suivante :

$$s(q,d) = \frac{\overrightarrow{q} \odot \overrightarrow{d}}{\|\overrightarrow{q}\|.\|\overrightarrow{d}\|} \quad [8]$$

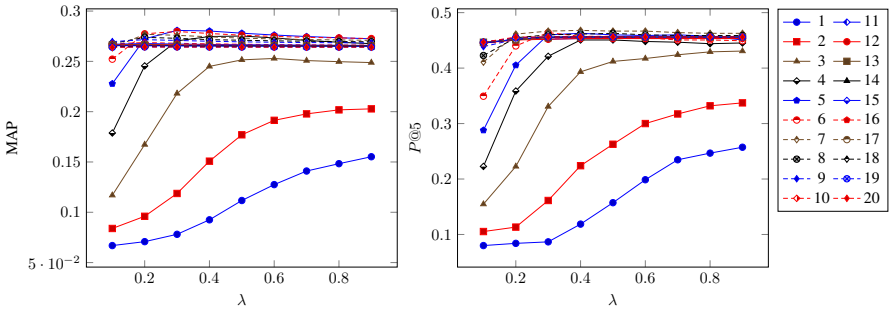
Avec $\odot$ est le produit scalaire entre les vecteurs $\overrightarrow{q}$ et $\overrightarrow{d}$. La valeur de similarité $s(q,d)$ représente le liens sémantique entre le terme q et le terme d.

Le paramètre α prend des valeurs dans $\{1, 2, ...20\}$ (au-delà de 20 les performances étaient stables), les paramètres λ, λ_1 et λ_2 prennent des valeurs dans $\{0.1, 0.2...0.9\}$. Rappelons que le paramètre α est utilisé pour contrôler l'impact de la similarité sémantique entre les termes (comme mentionné dans la section 3.1). Des expérimentations ont été menées sur la collection de données *AP 88-89* pour ajuster nos hyper-paramètres.
Nous avons utilisé l'outil trec_eval[2] pour calculer les valeurs MAP, $P@5$, $P@10$, $P@20$ et $nDCG@20$. Les analyses des équations 2 et 4 étaient similaires à celle de l'équation 3 pour laquelle nous décrivons l'impact des paramètres utilisés. La figure 1 montre l'évolution des performances, en termes de MAP et $P@5$, en utilisant le modèle BM25 pour calculer u_{score} dans l'équation 3. Nous remarquons que la MAP et la $P@5$ continuent d'augmenter jusqu'à $\lambda = 0.9$ pour $\alpha \in \{1, 2\}$. Pour $\alpha \in \{3, 4, 5, 6\}$ les performances sont plus stables avec $\lambda \in \{0.4, 0.5, 0.6\}$ puis diminuent lentement pour $\lambda > 0.6$. Pour $\alpha = 7$ les valeurs de MAP et de $P@5$ deviennent plus stables pour $\lambda \geq 0.4$.

Cette analyse montre l'impact du paramètre λ utilisé pour analyser l'influence des termes de la requête qui ne sont pas dans le document sur le processus d'appariement. Dans l'équation 3, $\lambda = 0.4$ donne les meilleurs résultats, expliquant l'importance des termes de la requête qui ne sont pas dans le document dans le processus d'appariement. Pour $\alpha \in \{1, 2, 3, 4\}$ les performances sont moins bonnes que celles pour $\alpha = 7$, car

1. https ://code.google.com/archive/p/word2vec/
2. http ://trec.nist.gov/trec_eval/

Figure 1 – Analyse de sensibilité aux paramètres α et λ dans la collection *AP 88-89*, pour l'équation 3 en utilisant BM25 dans u_{score}. Chaque courbe correspond à une valeur de α correspondante sur la légende à droite



la similarité sémantique avait plus d'impact sur le processus d'appariement. D'après (Zamani et Croft, 2017), puisque les représentations continues des mots sont basées sur la notion de contexte (Mikolov *et al.*, 2013b), elles peuvent conduire à l'utilisation des termes non pertinents dans l'appariement document-requête (exp : les termes *sécurisé* et *dangereux* ont une grande similitude sémantique mais *dangereux* n'est pas pertinent pour la requête *"transport de sécurité"*).

Grâce à l'analyse ci-dessus, nous avons défini les valeurs des paramètres qui correspondent au compromis entre la valeur de $P@5$ et la valeur de la MAP correspondante. Ce qui correspond à $\alpha = 7$ dans les deux équations 2 et 3 et $\lambda = 0.4$ dans l'équation 3. Pour l'équation 4, nous retenons la configuration suivante : $\alpha = 5$, $\lambda_1 = 0.5$ et $\lambda_2 = 0.3$.

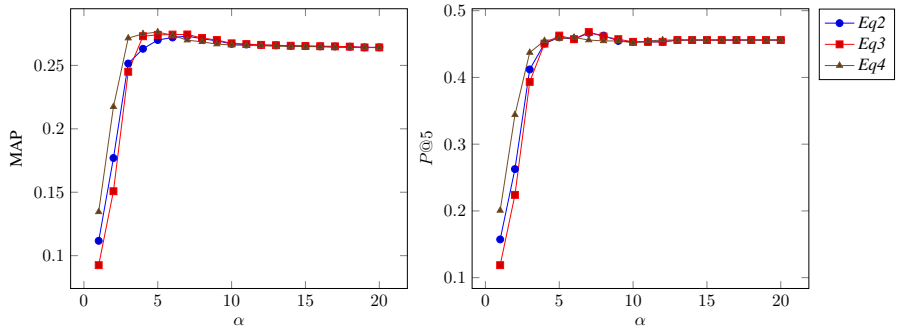
4.4. *Résultats et discussion*

Comme nous l'avons expliqué dans la section 3, nous analysons les contributions des termes de la requête qui sont absents dans le document au processus d'appariement sémantique par rapport aux modèles d'appariement exacts.

4.4.1. *Comportement des différentes stratégies d'appariement*

Nous avons d'abord analysé et comparé le comportement de chacune des stratégies d'appariement que nous avons définies, par rapport aux différentes valeurs du paramètre α utilisé pour contrôler l'impact de la similarité sémantique entre les termes. La figure 2 montre l'évolution des performances, en termes de MAP et de $P@5$ du processus d'appariement en utilisant les équations 2, 3 et 4 qui correspondent à *Eq2*, *Eq3* et *Eq4* respectivement sur cette figure. Cette analyse est réalisée sur la collection *AP 88-89*, avec $\lambda = 0.4$ dans *Eq3*, $\lambda_1 = 0.5$ et $\lambda_2 = 0.3$ dans *Eq4* et $\lambda_1 = 0.5$ et $\lambda_2 = 0.3$ dans *Eq4*. Selon les différentes courbes, nous pouvons remarquer que

Figure 2 – Comparaison de la performance, en termes de MAP et de $P@5$, des différentes stratégies d'appariement à l'aide des équations 2, 3 et 4 qui correspondent respectivement à *Eq2*, *Eq3* et *Eq4* sur chaque courbe, par rapport aux différentes valeurs du paramètre α dans la collection *AP 88-89*. Avec $\lambda = 0.4$ dans *Eq3*, $\lambda_1 = 0.5$ et $\lambda_2 = 0.3$ dans *Eq4* et $\lambda_1 = 0.5$ et $\lambda_2 = 0.3$ dans *Eq4*



toutes les équations se comportent de la même manière avec une légère différence des performances. La différence la plus importante est que pour $\alpha \in \{4, 5, 6, 7, 8\}$, les résultats des équations *Eq3* et *Eq4* sont légèrement meilleurs que les résultats de l'équation *Eq2*. Cependant, pour $\alpha > 8$ les performances de toutes les différentes stratégies diminuent du fait que l'apport de la similarité sémantique soit perdu. Ces résultats montrent d'une part que pour les différentes équations 2, 3 et 4, le paramètre α a le même apport est qui est de contrôler l'impact de la similarité sémantique entre les termes. D'une autre part, l'apport de la séparation entre les termes de la requête qui sont dans le document et ceux qui n'y sont pas (équations 3 et 4), a permis d'obtenir une certaine amélioration dans les valeurs de la MAP et la $P@5$ par rapport au traitement indifférencié de tout les termes de la requête (équation 2).

4.4.2. *Comparaison avec les modèles classiques*

Nous avons comparé aussi les différentes stratégies d'appariement par rapport aux modèles de RI classiques. Les tableaux 2 et 3 montrent les résultats obtenus sur deux collections de tests, Robust4 et GOV2, en utilisant respectivement BM25 et LM pour calculer la valeur de u_{score} dans chacune des équations 2, 3 et 4 et qui correspondent respectivement à *Eq2*, *Eq4* et *Eq4* dans les deux tableaux. Les résultats étiquetés avec $(+)$ ou $(-)$ montrent respectivement la significativité [3] des améliorations ou des baisses de performance par rapport au modèle de référence correspondant.
Tout d'abord, dans la collection Robust4, nous remarquons que dans les deux tableaux les stratégies d'appariement étudiées surpassent les modèles standard basés sur l'appariement exact, avec une amélioration significative des performances en termes de MAP et $P@5$ avec BM25 et en termes de $P@20$ avec ML. *Eq2* et *Eq3* surpassent les

3. Nous avons utilisé le test statistique *t-test* avec un niveau de confiance de 95%.

Tableau 2 – Résultats expérimentaux utilisant la BM25 en $u_{score}(.,.)$. Les résultats marqués de $+$ ou $-$ montrent respectivement la significativité des améliorations ou des baisses de performance par rapport au modèle BM25. Les valeurs maximales sont soulignées pour chaque ensemble de données et les valeurs en gras montrent l'amélioration des performances.

Collection	Model	MAP	P@5	P@10	P@20	nDCG@20
Robust4	*Eq2*	**0.2400**	**0.4851**	**0.4309**	**0.3548**	**0.4139**
	Eq3	**0.2403+**	**0.4851**	**0.4301**	**0.3550**	**0.4140**
	Eq4	**0.2401**	**0.4956+**	**0.4293**	**0.3772**	**0.4160**
	BM25	0.2362	0.4747	0.4285	0.3528	0.4105
GOV2	*Eq2*	0.2554	0.5141	0.5040	0.4893	0.4262
	Eq3	0.2553	0.5154	0.5013-	0.4893	0.4263-
	Eq4	0.2524	0.5128	0.4933	0.4829	0.4161
	BM25	0.2595	0.5463	0.5315	0.4977	0.4401

modèles de référence avec plus de 2% en termes de $P@5$ en utilisant le modèle BM25 et ML. *Eq4* surpassent les modèles de référence avec plus de 4% en termes de $P@5$ en utilisant BM25. Cependant, dans la collection GOV2, *Eq2* et *Eq3* ne surpassent les modèles classiques qu'en termes de $P@5$ lorsque le modèle ML est utilisé. Cependant, ces améliorations ne sont pas significatives.

Eq2 surpasse les modèles classiques dans la collection Robust4, mais ne montre pas d'améliorations significatives. Dans cette stratégie d'appariement, tous les termes du document et de la requête sont appariés sans distinction. *Eq3* et *Eq4* donnent de meilleurs résultats que *Eq2* lorsque BM25 est utilisé pour calculer u_{score} (Tableau 2). Dans *Eq3* et *Eq4*, les termes de la requête qui ne sont pas dans le document sont traités différemment par rapport à ceux qui y apparaissent. Ces résultats montrent l'impact de la distinction entre les termes de la requête en fonction de leur présence/absence dans le document et prouvent l'importance des termes de la requête qui ne figurent pas dans le document lors du processus de comparaison.

4.4.3. *Contributions des différents termes de la requête*

Afin d'expliquer les résultats obtenus dans chacune des collections utilisées par le processus d'appariement basé sur la présence/absence des termes de la requête dans les documents, nous avons analysé la contribution des différents termes de la requête au calcul du score des documents pertinents de ces collections. Soit $S_t(D,Q)$ le score de pertinence total calculé pour un document pertinent D par rapport à la requête Q par l'équation suivante :

$$S_t(D,Q) = S_a(D,Q) + S_p(D,Q) \quad [9]$$

Avec :
$S_a(D,Q) = \sum_{q_i \in Q,\ q_i \notin Q \cap D} \sum_{d_j \in D} sim(q_i, d_j)$ est le score calculé en fonction des termes de la requête qui sont absents dans le document,
$S_p(D,Q) = \sum_{q_i \in Q \cap D} \sum_{d_j \in D} sim(q_i, d_j)$ est le score calculé en fonction des

Tableau 3 – Résultats expérimentaux utilisant le ML en $u_{score}(.,.)$. Les résultats marqués de $+$ ou $-$ montrent respectivement la significativité des améliorations et des baisses de performance par rapport au modèle ML. Les valeurs maximales sont soulignées pour chaque ensemble de données et les valeurs en gras montrent l'amélioration des performances.

Collection	Model	MAP	P@5	P@10	P@20	nDCG@20
Robust4	*Eq2*	**0.2337**	**0.4378**	**0.4016**	**0.3392**	**0.3868**
	Eq3	**0.2336**	**0.4345**	**0.4016**	**0.3392**	**0.3865**
	Eq4	**0.2324**	0.4249	**0.3952**	**0.3438+**	**0.3856**
	ML	0.2310	0.4265	0.3936	0.3331	0.3807
GOV2	*Eq2*	0.2446	**0.4201**	0.4161	0.4077	0.3392
	Eq3	0.2444	**0.4201**	0.4161	0.4064	0.3384
	Eq4	0.2277-	0.3544	0.3638-	0.3695	0.2954-
	ML	0.2516	0.4054	0.4289	0.4094	0.3456

termes de la requête qui sont présents dans le document,
$sim(q_i, d_j)$ représente la similarité sémantique entre les termes q_i et d_j telle défini par l'équation 8.

La figure 3 montre les pourcentages de contribution des termes de la requête qui sont présents dans le document noté score S_p et des termes qui sont absents dans le document noté S_a au calcul du score total noté S_t pour les documents pertinents dans la collection *AP88-89* utilisée pour fixer les différents paramètres. Dans cette figure, chaque barre représentant un document pertinent dont le score est calculé par l'équation 9. Nous pouvons constater que les scores de la majorité des documents pertinents sont considérablement affectés par les similarités entres les termes de la requête qui sont absents dans les documents et les termes de ces documents. La figue 4 montre les pourcentages de contribution des différents termes des requêtes au calcul des scores des documents pertinents de chacune des collections GOV2 (figure 4a) et Robust4 (figure 4b).

Dans la figure 4, nous constatons que les scores des documents pertinents de la collection Robust4 (figure 4b) sont impactés par les similarités des termes de la requête qui ne sont pas dans le document (score S_a) de manière homogène. Ce qui explique l'amélioration des résultats de recherche obtenus par les différentes stratégies d'appariement étudiées dans ce papier (tableaux 2 et 3). Cependant, dans la collection GOV2 (figure 4a), les scores des documents pertinents ne sont pas influencés considérablement par les similarités entre les termes de la requête qui sont absents dans les documents pertinents et les termes de ces documents. De plus, dans la collection Robust4, prés de 12% des documents pertinents ne contiennent aucun terme de la requête, de ce fait, le traitement des termes de la requête qui sont absents dans les documents pertinents de manière différente a donné de bons résultats. Cependant dans la collection GOV2, on ne trouve que 1.1% des documents pertinents sans aucun terme de la re-

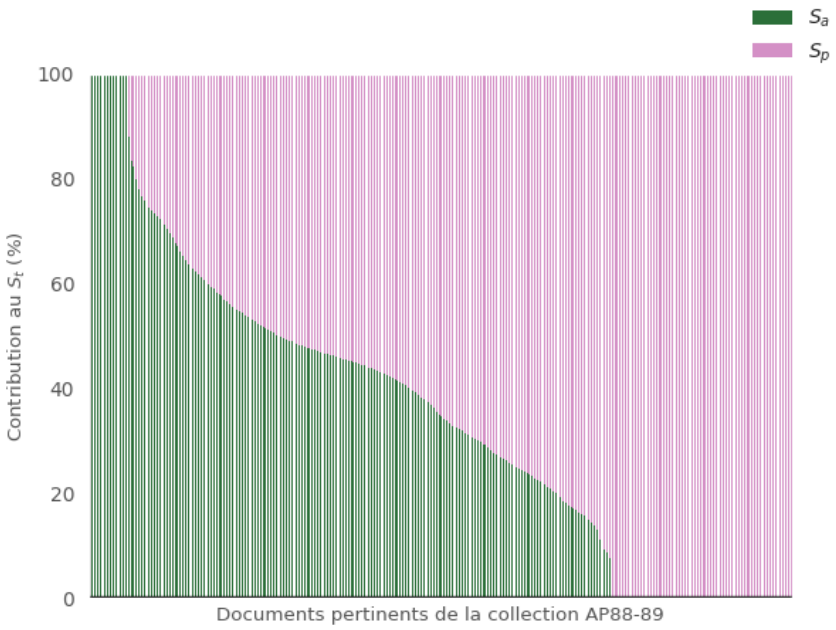
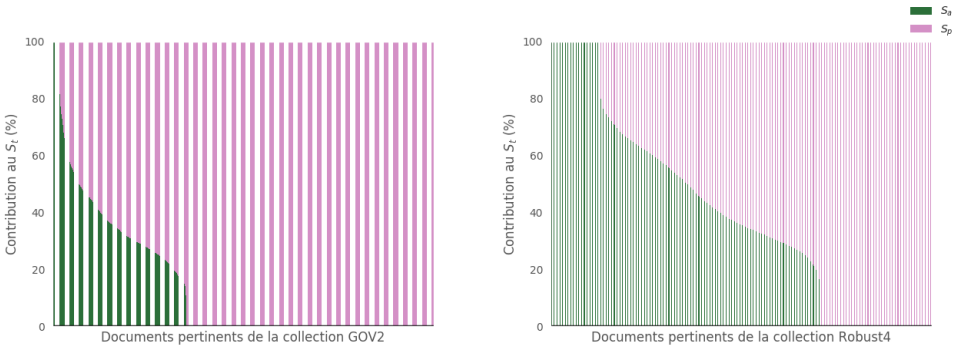


Figure 3 – Contributions des termes des requêtes en fonction de leur présence/absence dans le calcul du score des documents pertinents de la collection AP88-89, les barres représentant les documents pertinents de la collection.



(a) (b)

Figure 4 – Contribution des termes des requêtes en fonction de leur présence/absence dans les documents pertinents des collections GOV2 (a) et Robust4 (b), au calcul des valeurs de score. Les barres représentant les documents pertinents de chaque collection.

quête. Ce qui explique la non-influence de la séparation entre les termes de la requête par rapport à leur présence/absence dans les documents de la collection GOV2.

4.4.4. *Comparaison avec des modèles de l'état de l'art*

Dans le tableau 4, *avec-BM25* et *avec-LM* font référence à l'utilisation de BM25 et LM respectivement pour calculer u_{score} dans les équations *Eq2*, *Eq3* et *Eq4*. *RM-Cent* correspond au modèle de (Kuzi *et al.*, 2016) qui est un modèle d'expansion de requête utilisant le plongement lexical des mots. *NWT* fait référence au modèle de (Guo *et al.*, 2016), où tous les termes de la requête sont comparés indifféremment à tous les termes du document. Les tirets (-) renvoient à des résultats non disponibles[4].

Les stratégies d'appariement étudiées surpassent les modèles de l'état de l'art en termes de $P@5$ dans la collection Robust4 et en termes de $nDCG@20$ dans la collection GOV2. Dans toutes les stratégies d'appariement, nous exploitons toutes les interactions sémantiques entre les vecteurs des mots du document et les vecteurs des mots de la requête. Comparés aux modèles de l'état de l'art qui traitent indifféremment les termes de la requête : dans le modèle RM-Cent, les similarités sémantiques entre les mots sont utilisées pour sélectionner les termes les mieux adaptés pour l'expansion de la requête, puis la requête résultante est utilisée pour trier les documents avec un modèle de langue classique qui traite indifféremment les termes de la requête. Dans le modèle NWT, les vecteurs de mots sont utilisés pour capturer les similarités sémantiques entre les termes du document et tous les termes de la requête sans distinction. Bien que les stratégies d'appariement présentées dans le présent document ne soient toujours pas plus performantes que les modèles de référence, les résultats nous encouragent à faire davantage d'études en utilisant d'autres modèles de RI afin d'analyser la contribution des différents termes de la requête dans le processus d'appariement.

5. Conclusion et perspectives

Dans cet article, nous avons étudié l'appariement document-requête en fonction de la présence ou de l'absence des termes de la requête dans le document. Les similarités sémantiques entre les termes du document et les termes de la requête sont utilisées pour faire face à l'inadéquation du vocabulaire entre le document et la requête. Les résultats rapportés montrent que les termes de la requête qui ne figurent pas dans le document améliorent le processus d'appariement lorsqu'ils sont utilisés avec la similarité sémantique et permettent de donner de meilleurs résultats que les modèles classiques basés uniquement sur l'appariement exact. Les résultats obtenus sont comparables à ceux de certains modèles de l'état de l'art dans la collection Robust4. Cependant, en utilisant la collection GOV2 les différentes formules étudiées ne donnent pas une amélioration claire, du fait que les documents de cette collection sont

4. Dans ce tableau, nous rapportons les résultats des différentes approches de l'état de l'art tels présentés dans les articles correspondants pour *RM-Cent* (Kuzi *et al.*, 2016) et *NWT* (Guo *et al.*, 2016)

Tableau 4 – Comparaison du processus d'appariement étudié avec certains modèles de l'état de l'art. Les résultats soulignés dans chaque colonne représentent la valeur maximale par rapport aux différents modèles de référence.

Collection	Model		MAP	P@5	P@10	P@20	nDCG@20
Robust4	avec -BM25	*Eq2*	0.2400	0.4851	0.4309	0.3548	0.4139
		Eq3	0.2403	0.4851	0.4301	0.3550	0.4140
		Eq4	0.2401	0.4956	0.4293	0.3772	0.4160
	avec -LM	*Eq2*	0.2337	0.4378	0.4016	0.3392	0.3868
		Eq3	0.2336	0.4345	0.4016	0.3392	0.3865
		Eq4	0.2324	0.4249	0.3952	0.3438	0.3856
	RM-Cent		0.2910	0.4950	-	-	-
	NWT		0.2740	-	-	0.3800	0.4260
GOV2	avec -BM25	*Eq2*	0.2554	0.5141	0.5040	0.4893	0.4262
		Eq3	0.2553	0.5154	0.5013	0.4893	0.4263
		Eq4	0.2524	0.5128	0.4933	0.4829	0.4161
	avec -LM	*Eq2*	0.2446	0.4201	0.4161	0.4077	0.3392
		Eq3	0.2444	0.4201	0.4161	0.4064	0.3384
		Eq4	0.2277	0.3544	0.3638	0.3695	0.2954
	RM-Cent		0.3350	0.6230	-	-	-
	NWT		0.304	-	-	0.5240	0.4220

volumineux et la majorité d'entre eux contiennent tous les termes de la requête (ce qui est montré par l'analyse dans la section 4.4.3). Les travaux à venir concentreront notre réflexion sur la recherche des paramètres qui peuvent être utilisés pour rendre les méthodes d'appariement étudiées plus performantes sur de grandes collections de données comme GOV2.

6. Bibliographie

Dumais S. T., Furnas G. W., Landauer T. K., Deerwester S., Harshman R., « Using latent semantic analysis to improve access to textual information », *Proceedings of the SIGCHI conference on Human factors in computing systems*, Acm, p. 281-285, 1988.

Guo J., Fan Y., Ai Q., W. B., « Semantic Matching by Non-Linear Word Transportation for Information Retrieval », *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, p. 701-710, 2016.

Hofmann T., « Probabilistic latent semantic analysis », *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc., p. 289-296, 1999.

Jones K. S., Walker S., Robertson S. E., « A probabilistic model of information retrieval : development and comparative experiments : Part 2 », *Information processing & management*, vol. 36, n^{o} 6, p. 809-840, 2000.

Kenter T., De Rijke M., « Short text similarity with word embeddings », *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, ACM, p. 1411-1420, 2015.

Kuzi S., Shtok A., Kurland O., « Query expansion using word embeddings », *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, p. 1929-1932, 2016.

Lv Y., Zhai C., « Lower-bounding term frequency normalization », *Proceedings of the 20th ACM international conference on Information and knowledge management*, ACM, p. 7-16, 2011.

Metzler D., Bruce W., « Combining the language model and inference network approaches to retrieval », *Information processing & management*, vol. 40, nº 5, p. 735-750, 2004.

Mikolov T., Chen K., Corrado G. S., Dean J., « Efficient Estimation of Word Representations in Vector Space », 2013a.

Mikolov T., Sutskever I., Chen K., Corrado G. S., Dean J., « Distributed representations of words and phrases and their compositionality », *Advances in neural information processing systems*, p. 3111-3119, 2013b.

Pennington J., Socher R., Manning C. D., « Glove : Global Vectors for Word Representation. », *EMNLP*, vol. 14, p. 1532-1543, 2014.

Robertson S. E., Walker S., Jones S., Hancock-Beaulieu M. M., Gatford M. *et al.*, « Okapi at TREC-3 », *Nist Special Publication Sp*, vol. 109, p. 109, 1995.

Roy D., Ganguly D., Mitra M., Jones G. J., « Representing Documents and Queries as Sets of Word Embedded Vectors for Information Retrieval », 2016.

Zamani H., Croft W. B., « Embedding-based query language models », *Proceedings of the 2016 ACM international conference on the theory of information retrieval*, ACM, p. 147-156, 2016.

Zamani H., Croft W. B., « Relevance-based Word Embedding », *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, ACM, p. 505-514, 2017.