Extension du modèle de langue pour la RI avec la position du terme

Arezki Hammache[1], Mohand Boughanem[2]

1. Laboratoire LARI, Département d'informatique, Université Mouloud Mammeri
15000 Tizi-Ouzou, Algérie
arezki20002002@yahoo.fr

2. Laboratoire IRIT, Université Paul Sabatier
118 route de Narbonne 31062 Toulouse Cedex 09, France.
bougha@irit.fr

RESUME. *La plupart des modèles de RI se basent généralement sur la combinaison de trois facteurs dans leur fonction de pondération, qui sont : la fréquence du terme dans le document (TF), la fréquence du terme dans la collection (ou l'IDF) et la taille du document. Quelques approches ont proposé d'intégrer la position du terme dans le document dans l'objectif de surpondérer les termes qui apparaissent au début du document. Dans cet article, nous nous situons dans cette perspective. Précisément, nous proposons deux nouvelles techniques d'estimation du poids d'un terme en se basant sur ses positions dans le document. La première technique considère uniquement la positon de la première apparition du terme dans le document; la seconde technique prend en compte toutes les positions du terme dans le document. Nous avons ensuite intégré les facteurs obtenus dans un modèle de langue pour la RI. Deux techniques de lissage sont considérées dans ce modèle de langue: Dirichlet et Jelinek-Mercer. Les résultats expérimentaux obtenus sur deux collections de test TREC, montrent que notre modèle améliore significativement les deux modèles de langue de base: Dirichlet et Jelinek-Mercer. Notre modèle surpasse aussi un modèle de l'état de l'art, qui est le modèle CTR, basé sur la position du terme dans le document.*

ABSTRACT. *The weighting function of most IR models is usually based on a combination of three factors: Term Frequency (TF), Inverse Document Frequency (IDF) and document length. Some approaches have integrated term position in a document to boost the weight of terms appearing in the beginning of the document. In this article, we are in this perspective. Precisely, we propose two ways for estimating "position weighting". The first one exploits only the position of the first appearance of a term in a document; the second one considers all term positions in a document. We then integrated these factors into two smoothing methods from language model: Dirichlet and Jelinek-Mercer. Experiments conducted on two TREC test collections show that our model achieves a significant improvement over Dirichlet and Jelinek-Mercer models. Our model also outperforms state-of-the-art model, which is the Chronological Term Rank (CTR), based on term position in a document.*

Mots-clés : Position du terme, Modèle de langue pour la RI, Pondération.

KEYWORDS: *Term position, Language model for information retrieval, Term weighting.*

1. Introduction

La pondération des termes est l'une des fonctions principales en Recherche d'Information (RI). Elle consiste à assigner des poids aux termes de la requête dans les documents, dans l'objectif de les classer vis-à-vis de cette requête. La plupart des modèles de RI utilisent la combinaison de trois facteurs dans leur fonction de pondération. Ces facteurs sont : la fréquence du terme dans le document (*Term Frequency*), la fréquence du terme dans la collection ou l'*IDF* (*Inverse document Frequency*) et la taille du document.

Ces trois facteurs sont principalement basés sur la représentation en sac de mots des documents. Cette représentation facilite grandement les calculs. Cependant, elle ignore l'ordre d'apparition des termes dans le document, ce qui engendre le problème d'ambiguïté (polysémie). L'une des pistes poursuivies pour aller au delà de cette représentation est l'utilisation des positions des termes dans le document.

Trois principales directions ont été investies pour introduire la position du terme dans le processus de correspondance document-requête. La première utilise la structure du document (Oglivie et Callan, 2003) (Robertson *et al.,* 2004), et cela en affectant plus de poids aux termes qui apparaissent dans certaines parties du document, comme le titre. La seconde est basée sur la proximité des termes de la requête dans le document (Metzler et Croft, 2005) (Lv et Zhai, 2009) (Hammache *et al.*, 2014) (Lioma *et al.,* 2015). Ainsi, un bon document est celui qui contient les termes de la requête les uns proches des autres. La troisième direction est basée sur l'utilisation de la position exacte du terme dans le document, plus exactement pour surpondérer les termes qui apparaissent au début du document. Troy and al ont introduit un modèle nommé CTR (Chronological Term Rank) (Troy et Zhang, 2007). L'objectif de ce modèle est d'étendre le modèle BM25 afin de surpondérer les termes qui apparaissent au début du document. Seo et al ont proposé une méthode d'apprentissage pour le filtrage des documents non-pertinents et qui sont bien classés (Seo et Jeon, 2009). L'une des caractéristiques utilisée par cette méthode est la position des phrases pertinentes dans le document. Notre modèle s'inscrit dans cette direction.

Notre modèle a des similarités avec le modèle CTR (Troy et Zhang, 2007), néanmoins notre modèle présente des différences significatives. Notamment, au lieu d'utiliser uniquement la position de la première apparition du terme dans le document, notre modèle prend en compte aussi toutes les positions du terme dans le document. De plus, l'estimation du poids de la position du terme dans le document est réalisée d'une manière plus formelle et intégrée dans le modèle de langue pour la RI, plutôt que dans un modèle probabiliste classique comme c'est le cas dans le modèle CTR.

Nous proposons dans ce papier un modèle de langue mixte qui combine le modèle de langue de base du document et un modèle de document basé sur les positions du terme. Pour cela, nous avons utilisé deux techniques de lissage : Jelinek-Mercer et Dirichlet. Nous avons évalué notre modèle sur deux collections de test TREC. Les résultats expérimentaux obtenus montrent que notre modèle

améliore significativement les deux modèles de langue de base: Jelinek-Mercer et Dirichlet. Notre modèle améliore aussi un des modèles de l'état de l'art à savoir le modèle Chronological Term Rank (CTR) (Troy et Zhang, 2007).

Le reste de ce papier est organisé comme suit: la section 2 présente les travaux connexes. La section 3 présente notre modèle de langue intégrant les positions du terme dans le document. Nous rapportons les résultats expérimentaux dans la section 4. Enfin, dans la section 5, nous concluons notre travail et énumérons quelques perspectives.

2. Etat de l'art

L'utilisation de la position du terme dans le document a montré son efficacité dans plusieurs tâches de Traitement Automatique de Langue Naturelle (TALN) (Hulth, 2003) (Xue et Zhou, 2006) et de Recherche d'Information (Oglivie et Callan, 2003) (Metzler et Croft, 2005) (Troy et Zhang, 2007) (Zhao et Yun , 2009) (Lv et Zhai, 2009) (Hammache *et al.*, 2014) (Lioma *et al.,* 2015).

La première utilisation de la position du terme dans le document est réalisée par Luhn (Luhn, 1958). Il a utilisé dans son approche la position et la fréquence du terme dans la phrase, pour mesurer l'importance de cette dernière dans le document. Les phrases les plus importantes sont ensuite utilisées pour construire le résumé du document.

En TALN, plusieurs travaux ont exploité la position du terme dans le document pour diverses finalités : extraction des mots-clés (Hulth, 2003) et catégorisation des textes (Xue et Zhou, 2006).

Dans le domaine de la RI plusieurs travaux ont introduit la position du terme dans le document selon trois points de vues différents : la structure du document (Oglivie et Callan, 2003) (Robertson *et al.,*2004) (Kim et Croft, 2012) (Zamani, *et al.*, 2018), la proximité des termes de la requête (Metzler et Croft, 2005) (Lv et Zhai, 2009) (Hammache *et al.*, 2014) (Lioma *et al.,* 2015) et la position exacte du terme dans le document (Troy et Zhang, 2007).

L'utilisation de la structure du document comme source d'évidence consiste à pondérer différemment les parties du document. Par exemple, la partie titre du document est généralement surpondérée. Ainsi, les termes qui apparaissent dans les parties surpondérées seront par conséquent surpondérés. Dans (Oglivie et Callan, 2003), un modèle de langue mixte a été proposé dans le contexte de deux tâches de recherche de documents HTML: recherche de page d'accueil et recherche de page nommée. Ce modèle combine six représentations du document. Les représentations de texte intégral et de titre sont celles qui ont donné les meilleures performances pour les deux tâches considérées. La différence entre ces approches, qui utilisent la structure du document, et la notre réside dans le fait que notre approche de pondération considère la(es) position(s) exacte(s) du terme dans le document et non pas la (es) partie(s) dans laquelle (lesquelles) il apparait.

La seconde utilisation de la position du terme dans le document, consiste d'abord à mesurer la proximité des termes de la requête dans le document, ensuite à intégrer ce facteur de proximité dans le calcul de la similarité document-requête (Lv et Zhai, 2009). Lv et Zhai (Lv et Zhai, 2009) ont proposé un modèle de langue de position, dans lequel un modèle de langue pour chaque position du document est construit, nommé PLM pour "Positional Laguage Model". Le score final du document est alors obtenu par la combinaison des PLMs des termes de la requête. Cependant, ce modèle ignore l'importance des positions exactes des termes de la requête dans le document.
D'autres travaux ont intégré la proximité des termes sous un autre angle: par l'utilisation d'unités plus complexes telles que les phrases et les termes composés. Précisément, ces approches considèrent que la requête (le document) a deux représentations distinctes. La première est basée sur les termes simples (uni-gramme) et la seconde est basée sur des unités composées : bi-grammes (Song et Croft, 199) (Srikanth et Srihari, 2002), n-grammes (Metzler et Croft, 2005), termes composés (Hammache *et al.*, 2014) (Lioma *et al.,* 2015). Ces deux représentations sont ensuite combinées pour classer les documents vis-à-vis de la requête.
L'un des problèmes rencontrés par ces approches, qui exploitent la proximité des termes de la requête, est le temps d'exécution élevé de la requête. Ceci est dû au fait que le calcul du facteur de proximité est réalisé en ligne et utilise les positions de tous les termes de la requête. Dans notre approche, par contre, nous utilisons uniquement les positions d'un terme de la requête dans le document, indépendamment des positions des autres termes de la requête.

La troisième direction d'exploitation de la position du terme dans le document, consiste à utiliser la position exacte du terme dans le document comme facteur additionnel de la pondération. Peu de travaux ont été réalisés dans cette perspective. L'idée principale de cette catégorie d'approches est de surpondérer les termes qui apparaissent au début du document. Dans (Troy et Zhang, 2007) Troy et al ont introduit une extension du modèle BM25 par l'utilisation de la position exacte du terme dans le document. Cette extension exploite le classement chronologique des termes (Chronological Term Rank : CTR), qui a pour objectif de surpondérer les termes qui apparaissent au début du document. Une série de fonctions empiriques ont été proposées et utilisées pour formaliser ce facteur additionnel (CTR). Le modèle CTR a montré des améliorations significatives par rapport aux modèles de l'état de l'art sur plusieurs collections de test TREC (Troy et Zhang, 2007). Dans (Acun *et al*, 2013), le modèle CTR est appliqué avec succès dans le domaine de détection et de suivi de sujets (Topic Detection and Tracking : TDT).
Seo et al (Seo et Jeon, 2009) ont proposé une méthode de filtrage des documents non-pertinents, qui contiennent beaucoup de termes de la requête. Cette méthode procède en deux étapes. La première étape consiste à "mapper" le score des phrases et leur position dans le document pour obtenir un graphique nommé "Relevance-Flow Graph". Dans la seconde étape un modèle de régression logistique est utilisé pour la prédiction de la pertinence du document en se basant sur son "Relevance-Flow Graph". L'une des caractéristiques utilisée par ce modèle d'apprentissage est la position du premier pic du graphique. Cette caractéristique repose sur l'intuition suivante: "un document pertinent est susceptible de contenir des phrases pertinentes

au début du document". Enfin, la pertinence prédite a été utilisée pour reclasser la liste initiale des documents retournés. Les résultats obtenus ont montré une amélioration significative par rapport aux résultats initiaux (Seo et Jeon, 2009).

Notre modèle a des similarités avec le modèle CTR (Troy et Zhang, 2007) dans le sens où nous intégrons aussi la position exacte du terme pour surpondérer les termes qui apparaissent au début du document. Cependant, notre modèle présente certaines différences et avantages.

Premièrement, en plus de l'exploitation de la position de la première apparition du terme dans le document, comme c'est le cas dans le modèle CTR, nous proposons un deuxième cas de figure qui exploite toutes les positions du terme dans le document, car nous considérons que le premier cas ne permet pas de bien capturer toute l'importance de ce facteur (position du terme).

Deuxièmement, nous formalisons ce facteur avec une méthode plus systématique que celles employées par le modèle CTR. Précisément, nous avons fait le choix d'utiliser la fonction gaussienne pour modéliser la distribution des poids sur les positions du document. Ce choix est dicté par le fait que cette fonction a montré son efficacité dans plusieurs travaux antérieurs (Lv et Zhai, 2009) (Gerani *et al*, 2010) (Kacem *et al*, 2017) basés sur la position du terme.

Troisièmement, nous avons estimé un modèle de document basé sur les positions du terme dans le document. Nous l'avons ensuite intégré dans le modèle de langue pour la recherche d'information, plutôt que dans un modèle probabiliste classique, comme c'est le cas dans le modèle CTR. Ainsi, nous proposons un modèle de langue mixte qui combine le modèle de langue de base du document avec un modèle de document basé sur les postions du terme dans le document.

3. Extension du modèle de langue pour la RI avec la position du terme

La représentation en sac de mots des documents est largement utilisée en recherche d'information car elle est simple à mettre en œuvre. Cette représentation ignore la position des termes dans le document, ce qui ne permet pas de bien capturer la sémantique du document.

Dans notre approche nous prenons en compte la position du terme dans le document sous l'hypothèse suivante : "les termes qui apparaissent au début du document sont les meilleurs représentants du contenu du document, par conséquent, ils doivent être surpondérés". L'importance du terme diminue progressivement en allant vers la fin du document. Afin de mettre en œuvre cette hypothèse, nous proposons deux cas distincts: dans le premier cas on considère uniquement la position de la première apparition du terme dans le document, comme cela est fait dans (Troy et Zhang, 2007). Dans le second cas, on considère toutes les positions du terme dans le document. Nous formalisons ces deux cas sous le modèle de langue.

Dans ce qui suit, nous présentons d'abord le modèle de document basé sur les positions du terme. Nous détaillons, ensuite, l'intégration de ce modèle dans le modèle de langue de base.

3.1 Modèle de document basé sur les positions du terme

Dans notre modèle, nous considérons une requête Q et un document D représentés avec le vocabulaire suivant : $V = \{t_1 \dots t_i \dots t_n\}$.

Nous assumons qu'un document D est représenté comme un ensemble de triplets, comme suit :

$$D = \{(t_1;\ tf(t_1);\ P^{t_1}), \dots, (t_i;\ tf(t_i);\ P^{t_i}), \dots, (t_n;\ tf(t_n);\ P^{t_n})\}$$

Chaque triplet contient: l'intitulé du terme (t), la fréquence du terme dans le document $(tf(t))$ et les positions absolues du terme dans le document $P^t =< p_1^t, \dots, p_j^t, \dots, p_{tf(t)}^t >$.

En se basant sur cette représentation, nous assignons un poids $(W(t, D, p^t))$ pour chaque position p^t du terme t dans le document D. Précisément, nous attribuons un poids élevé aux premières positions du document et ce poids diminue progressivement vers la fin du document. Pour répondre à cette exigence, nous proposons d'utiliser la fonction gaussien pour estimer le poids du terme t apparaissant à la position p^t dans le document D comme suit:

$$W(t, D, p^t) = e^{-\frac{1}{2}\delta\left(\frac{p^t}{|D|}\right)^2} \quad (1)$$

Où $W(t, D, p^t)$ est le poids de la position p^t du terme t dans le document D, $|D|$ est la taille du document et δ est un paramètre pour contrôler la dispersion du "poids de la position" dans le document. Plus δ est petit plus la courbe de la distribution des poids sur les positions du document devient moins évasée.

Nous avons défini deux cas pour l'estimation du poids final d'un terme (noté $W(t, D, P^t)$), basé sur ses positions dans le document.

Le premier cas prend en compte uniquement la positon de la première apparition du terme dans le document (noté $W_{First}(t, D, P^t)$). Nous formalisons ce poids comme suit:

$$W_{First}(t, D, P^t) = W(t, D, p_1^t) \quad (2)$$

Où p_1^t est la position de la première apparition du terme t dans le document D.

Le second cas prend en compte toutes les positions du terme dans le document (noté $W_{All}(t, D, P^t)$). Nous le formalisons comme suit :

$$W_{All}(t, D, P^t) = \sum_{k=1}^{tf(t)} W(t, D, p_k^t) \quad (3)$$

Afin d'obtenir un modèle de document basé sur les positions du terme (une probabilité) nous normalisons les poids. Le modèle est noté $(P_{Pos}(t, D))$, il est formalisé comme suit :

$$P_{Pos}(t, D) = \frac{W(t, D, P^t)}{\sum_{\forall t_j \in D} W(t_j, D, P^{t_j})} \quad (4)$$

Nous notons respectivement $P_{PosFirst}(t, D)$ et $P_{PosAll}(t, D)$ les modèles du document pour les deux cas identifiés précédemment.

3.2 Extension du modèle de lange de base avec le modèle de document basé sur la position du terme

Le modèle de langue pour la recherche d'information est un modèle probabiliste (Ponte *et al.,* 1998). L'idée de base de ce modèle consiste d'abord à construire (estimer) le modèle de langue pour chaque document; ensuite à classer les documents selon la probabilité de générer la requête à partir de leur modèles. Cette probabilité est exprimée comme suit :

$$P(Q|D) = \prod_{t \in Q} P(t|D) \quad (5)$$

La probabilité $P(t|D)$ est estimée comme suit :

$$P(t|D) = \lambda P_{ML}(t|D) + (1 - \lambda) P_{ML}(t|C) \quad (6)$$

Où $P_{ML}(t|D)$ et $P_{ML}(t|C)$ sont, respectivement, les estimations par maximum de vraisemblance du terme t dans le document D et dans la collection C et λ est un paramètre de lissage. Deux principales méthodes de lissage sont utilisées : Jelinek-Mercer (JM) et Dirichlet priors (Dir) (Zhai et Lafferty, 2004). Dans la méthode JM le paramètre de lissage λ est une constante qui prend ses valeurs dans l'intervalle [0,1]. Dans la seconde méthode (Dir) le paramètre dépend de la taille du document, il est exprimé comme suit : $\lambda = \frac{|D|}{|D|+\mu}$. Où $|D|$ est la taille du document et μ est hyper paramètre.

Nous avons combiné le modèle de document basé sur les positions du terme, défini par la formule (4), avec le modèle de langue de base décrit par la formule (6). Ceci est réalisé par le lissage de la probabilité du terme dans le document $P_{ML}(t|D)$ par sa probabilité basé sur ses positions dans le document $P_{Pos}(t|D) = P_{Pos}(t, D)$. Ainsi nous exprimons le modèle obtenu (noté $P_{Ml\_Pos}(t|D)$) comme suit :

$$P_{ML\_Pos}(t|D) = \lambda\big((1 - \alpha) \times P_{ML}(t|D) + \alpha \times P_{Pos}(t|D)\big) + (1 - \lambda) P_{ML}(t|C) \quad (7)$$

Où α est un paramètre qui contrôle l'apport de modèle de document basé sur les positions du terme. Nous avons adopté cette méthode d'incorporation car nous combinons deux vues sur un même document. Nous avons ensuite combiné le modèle de document obtenu avec le modèle de la collection $P_{ML}(t|C)$.

4. Expérimentations et résultats

4.1. Collections de test et configuration expérimentale

Nous avons implémenté notre modèle sous la plate forme Terrier (Macdonald et He, 2008). Les expérimentations ont été réalisées sur deux collections de test TREC:

la collection AP88 (Associated Press, 1988), qui est une collection homogène et la collection WT10G, qui est une collection web, plus grande et hétérogène. En plus, du facteur taille des collections, nous avons choisi d'utiliser ces deux collections, car leur style de formatage est totalement différent. La collection AP88 a un style de formatage assez simpliste, où la grande partie du contenu des documents se trouve à l'intérieur des balises <Text>. Cependant, la collection WT10G a un style de formatage riche, qui inclut les différentes balises HTML. Ainsi, le contenu des documents est distribué sur les différentes parties du document.
Les deux collections de test ont été indexées, où les termes vides sont éliminés et l'algorithme de Porter (Porter, 1968) est utilisé.

Afin d'étudier le comportement de notre modèle, nous avons utilisé deux types de requêtes: des requêtes courtes, dans lesquelles seule la partie titre est utilisée et des requêtes longues où les parties: titre, description et narrative, sont utilisés.

La table 1 ci-dessous montre quelques statistiques sur les collections et les requêtes utilisées.

Table 1. Statistiques sur les collections et les requêtes utilisées

Collection	#documents	Requêtes d'apprentissage	Requêtes de test
AP88	79,919	51–100	101–150
WT10G	1,692,096	451–500	501–550

4.2 Valeurs des paramètres

Pour trouver les valeurs optimales des paramètres de notre modèle et ceux des modèles Baseline, afin d'optimiser la valeur de *MAP*, nous avons utilisé 50 requêtes d'apprentissage pour chaque collection de test. Une fois que les valeurs des paramètres sont apprises, nous les appliquons sur les 50 requêtes restantes.

Les modèles de base utilisés sont : le modèle Dirichlet et le modèle Jelinek-Mercer. Le modèle Dirichlet, noté *Dir*, a un seul paramètre μ. Pour estimer sa valeur optimale nous avons varié sa valeur de 100 à 5000 avec un pas de 100. Le modèle Jelinek-Merce, noté JM, a aussi un seul paramètre λ. Pour estimer sa valeur optimale nous avons varié sa valeur de 0.1 à 0.9 avec un pas de 0.05.

Nous avons considéré deux configurations dans notre approche. La première configuration est basée sur l'utilisation de la position de la première apparition du terme dans le document, notée *PosFirst*. La seconde configuration utilise toutes les positions du terme dans le document, notée *PosAll*. Nous avons introduit les facteurs obtenus de ces deux configurations dans deux modèles de langue: Dirichlet et Jelinek-Mercer. L'extension de ces deux modèles (*Dir* et *JM*) avec les deux configurations (*PosFirst* et *PosAll)* sont notées respectivement : *Dir-PosFirst, Dir-PosAll* et *JM-PosFirst, JM-PosAll*.

Notre modèle a deux paramètres de contrôle à fixer expérimentalement, α et δ. Ces paramètres sont des constantes qui prennent leur valeur, respectivement, dans

l'intervalle $\alpha\epsilon[0,1]$ avec un pas de 0,1 et $\delta\epsilon[0,1]$ avec un pas de 0.005. Pour définir les valeurs des deux paramètres, nous fixons d'abord la valeur du paramètre δ, puis nous varions la valeur du paramètre α.

La table 2. ci-dessous présente les valeurs des paramètres ayant obtenu les meilleures performances pour les différents modèles, en fonction du type de requêtes utilisées (courte et longue).

Table 2. Valeurs des paramètres utilisés dans les différents modèles

Collection	Modèle (Paramètres) / Type requête	Dir (μ)	Dir-PosFirst (δ,α)	Dir-PosAll (δ,α)	JM (λ)	JM-PosFirst (δ,α)	JM-PosAll (δ,α)
AP88	Courte	500	(0.1,0.2)	(0.1,0.2)	0.75	(0.05,0.3)	(0.1,0.3)
	Longue	500	(0.05,0.2)	(0.05,0.4)	0.9	(0.05,0.5)	(0.1,0.3)
WT10G	Courte	2000	(0.2,0.2)	(0.2,0.2)	0.15	(0.025,0.4)	(0.05,0.4)
	Longue	400	(0.025,0.2)	(0.025,0.2)	0.8	(0.025,0.2)	(0.05,0.3)

Afin d'évaluer les différents modèles, nous avons utilisé trois métriques: la MAP (Mean Average Precision) et les Précisions à N documents (P@10 et P@20).

Afin de vérifier la significativité des résultats par rapport aux modèles de base (Dir et JM), nous avons effectué le test de Wilcoxon et nous avons marqué les résultats à l'aide du symbole "$^{+}$" lorsque le test passe le niveau de confiance de 95%. Les meilleurs résultats sont mis en gras.

4.3 Evaluation

Nous présentons dans cette section les résultats de l'évaluation des deux configurations (PosFirst et PosAll) par rapport aux modèles de langue de base (Dir et JM). Les tables 3. et 4. présentent les résultats de cette évaluation avec les requêtes courtes et longues.

Table 3. Résultats des différents modèles avec les requêtes courtes

Collection	Modèle		MAP	P@10	P@20
AP88	Dir		0.2429	0.3408	0.2969
	Dir	PosAll	**0.2544**$^{+}$	**0.3533**$^{+}$	**0.3081**$^{+}$
		PosFirst	0.2525^{+}	0.3449	0.3020^{+}
	JM		0.2281	0.3265	0.2735
	JM	PosAll	**0.2474**$^{+}$	**0.3286**	**0.2888**$^{+}$
		PosFirst	0.2432^{+}	0.3265	0.2837^{+}
WT10G	Dir		0.2094	0.3420	0.3040
	Dir	PosAll	0.2145^{+}	**0.3800**$^{+}$	**0.3391**$^{+}$
		PosFirst	**0.2164**$^{+}$	0.3794^{+}	0.3378^{+}
	JM		0.1504	0.2640	0.2360
	JM	PosAll	**0.1707**$^{+}$	**0.3120**$^{+}$	0.2650^{+}
		PosFirst	0.1650^{+}	0.3080^{+}	**0.2680**$^{+}$

Table 4. Résultats des différents modèles avec les requêtes longues

Collection	Modèle		MAP	P@10	P@20
AP88	Dir		0.2967	0.3796	0.3520
	Dir	PosAll	**0.3238**$^{+}$	**0.4061**$^{+}$	**0.3735**$^{+}$
		PosFirst	0.3191^{+}	**0.4061**$^{+}$	0.3622^{+}
	JM		0.3123	0.3959	0.3612
	JM	PosAll	**0.3277**$^{+}$	**0.4102**$^{+}$	**0.3673**$^{+}$
		PosFirst	0.3232^{+}	0.4082^{+}	0.3663
WT10G	Dir		0.1023	0.2440	0.1860
	Dir	PosAll	**0.1196**$^{+}$	**0.2560**$^{+}$	**0.2130**$^{+}$
		PosFirst	0.1165^{+}	0.2540^{+}	0.2042^{+}
	JM		0.1067	0.2280	0.1890
	JM	PosAll	**0.1257**$^{+}$	**0.2560**$^{+}$	**0.2040**$^{+}$
		PosFirst	0.1226^{+}	0.2500^{+}	0.1990^{+}

L'analyse globale de nos résultats montre que notre modèle, avec ses deux configurations, obtient de bonnes performances, par rapport aux modèles de langue de base (Dirichlet et Jelinek-Mercer) et cela sur les deux collections utilisées. Cela montre que la surpondération des termes qui apparaissent au début du document peut être utile pour la RI.

De plus, la version PosAll obtient de meilleurs résultats que la version PosFirst. Cela indique que la prise en compte de toutes les positions du terme dans le document, est plus appropriée pour modéliser la pondération des positions du terme.

Nous analysons en détail ci-dessous le comportement et l'impact de notre modèle suivant deux axes: (1) type de la collection utilisée (collection homogène AP88 et collection Web hétérogène WT10G) et (2) type du modèle étendu (Dir ou JM).

4.3.1 Impact selon le type de la collection

Les résultats obtenus avec toutes les mesures d'évaluation utilisées, MAP, P@10 et P@20, sur la collection hétérogène "WT10G", ayant un style de mise en forme riche, sont globalement supérieurs à ceux obtenus sur la collection homogène "AP88", ayant un style de mise en forme simple. Précisément, les meilleures améliorations obtenues en termes de MAP, de P@10 et de P@20 sur les collections "AP88" et "WT10G" sont respectivement: (+9.13%, +6.98% et +6.11%) et (+17.81%, +18,18% et +14,52%). Cela est dû, probablement, au fait que les auteurs des documents Web tendent à concentrer leurs principales idées dans les premières parties du document. Cependant, les auteurs des documents de presse "AP88" ne concentrent pas trop leur idée sur les premières parties d'un document. Cela nous porte à croire que notre approche peut être utilisée avec succès dans le domaine de la recherche d'information sur le Web.

La version PosAll fonctionne mieux que la version PosFirst avec la plupart des mesures d'évaluation utilisées, MAP, P@10 et P@20, et ce quelle que soit la collection considérée. Sur la collection "AP88", les meilleures améliorations

obtenues en termes de MAP, P@10 et P@20 avec les versions PosFirst et PosAll sont respectivement: (+7,55%, +6,98% et +3,73%) et (+9,13%, +6,98 % et +6,11%). Les meilleures améliorations obtenues sur la collection "WT10G", avec les versions PosFirst et PosAll sont respectivement: (+14,90%, +16,66% et +13,55%) et (+17,81%, +18,18% et +14,52%). Cela confirme que notre hypothèse de prise en compte de toutes les positions du terme dans la fonction de pondération est plus appropriée que celle qui considère uniquement la position de la première apparition du terme dans le document, comme cela est fait dans (Troy et Zhang, 2007), et ce, quelle que soit la collection utilisée.

4.3.2 Impact selon le type du modèle étendu

L'analyse des résultats obtenus en étendant les deux modèles (Dir et JM) montre que les deux modèles sont améliorés en termes de MAP, P@10 et P@20. Cependant, les améliorations obtenues sur l'extension du modèle JM sont légèrement meilleures que celles obtenues sur l'extension du modèle Dir, à l'exception de la collection AP88 avec des requêtes longues. La raison peut être expliquée par le fait que notre modèle compense, probablement, le manque de normalisation par la taille du document dans le modèle JM. Ainsi, les meilleures améliorations obtenues par rapport aux modèles Dir et JM sont respectivement: (+16,91%, +11,11% et +14,52%) et (+17,81%, +18,18% et +13,55%).

Nous notons également que sur les deux modèles, la version PosAll reste meilleure que la version PosFirst. Cela montre que notre hypothèse reste valable quel que soit le modèle étendu.

4.3.3 Exemple d'une requête

Afin d'avoir une vision plus précise de l'impact du modèle proposé, nous avons examiné manuellement certaines requêtes. Par exemple, la requête numéro 104, qui contient les termes: "**Catastrophic Health Insurance**" (Figure 1.), nos extensions du modèle Jelinek-Mercer avec les configurations PosFirst et PosAll obtiennent respectivement une précision moyenne de 0,5045 et 0,6369, tandis que le modèle Jelinek-Mercer obtient 0,3256 de précision moyenne.

<top>
<num> Number: 104
<title> Catastrophic Health Insurance
<desc> Description:
Document will enumerate provisions of the U.S. Catastrophic Health Insurance
Act of l988, or the political/legal fallout from that legislation.....
<narr> Narrative: .....
</top>

Figure 1. Requête 104

La table ci-dessous présente le classement des documents pertinents avec les trois modèles (JM, JM-PosFirst, JM-PosAll), en utilisant la partie titre de la requête uniquement. En outre, nous montrons également les positions des termes de la requête dans ces documents.

Table 5. Le classement des documents pertinents, sur les 1 000 documents renvoyés, avec les trois modèles de recherche (JM, JM-PosFirst et JM-PosAll) (*) non renvoyé

Document	Rang JM	Rang JM-PosFirst	Rang JM-PosAll	Taille document	Termes	Positions
AP880316-0154	2	10	8	226	catastroph	37;65;96;103;163;224
					health	14;46;66;116;154;175;207;225
					insur	164
AP880526-0030	4	6	2	173	catastroph	7;13;17;24;82;133
					health	18;25
					insur	26
AP880603-0052	5	5	1	204	catastroph	13;18;22;30;163
					health	14;19;23;31
					insur	32
AP880526-0027	10	1	3	472	catastroph	13;17;32;183,290
					health	14;18;33;72;184;190;291
					insur	19;34
AP880609-0025	17	4	6	350	catastroph	13;18;28;154
					Health	14;19;29;231;240
					insur	30
AP880602-0163	21	11	7	255	catastroph	12;29
					health	13;30;62
					insur	31
AP880608-0295	23	27	29	421	catastroph	140;155
					health	24;54;68;141;327;358
					insur	411;418
AP880609-0027	35	14	11	475	catastroph	13;21;35;146
					Health	22;38
					insur	39;43
AP880521-0022	39	13	15	354	catastroph	12;37
					health	13;38;199
					insur	39
AP880701-0160	42	34	25	384	catastroph	40;171
					Health	41;260;269
					insur	42
AP881117-0022	NR(*)	NR(*)	NR(*)	430	Health	348
					insur	59

Nous remarquons dans cet exemple que le rang de presque tous les documents pertinents a été promu par notre modèle avec ses deux versions, par rapport au modèle Jelinek-Mercer. De plus, la version PosAll est plus performante que la

version PosFirst. Par exemple, le document "AP880602-0163" passe du rang 21 avec le modèle Jelinek-Mercer au rang 11 avec le modèle JM-PosFirst et au rang 7 avec le modèle JM-PosAll. La première amélioration s'explique par le fait que les termes de la requête "**Catastrophic Health Insurance**" apparaissent pour la première fois au début du document; ils apparaissent respectivement aux positions suivantes : 12,13 et 31. La seconde amélioration s'explique par le fait que toutes les occurrences des termes de requête apparaissent au début du document, leurs positions sont respectivement : <12,29>, <13,30,62> et <31>.

4.4. Comparaison avec le modèle CTR

Dans cette section nous présentons la comparaison de notre modèle avec le modèle CTR (Troy et Zhang, 2007). Pour une comparaison équitable entre les deux modèles, nous avons suivi le même protocole que nous avons utilisé dans notre modèle pour l'estimation des paramètres du modèle CTR. Ainsi, nous avons utilisé 50 requêtes d'apprentissage sur chaque collection. Les valeurs des paramètres apprises sont ensuite utilisées pour tester le modèle CTR sur les 50 requêtes restantes.

Les tables ci-dessous montrent les résultats des deux modèles. Nous avons présenté uniquement, dans notre cas, le modèle qui a fournit les meilleurs résultats pour chaque type de requête: le modèle Dirichlet pour les requêtes courtes et le modèle Jelinek-Mercer pour les requêtes longues.

Table 6. Comparaison entre les différents modèles avec les requêtes courtes

Collection	Modèle	MAP	P@10	P@20
AP88	Dir	0,2429	0,3408	0,2969
	Dir-PosFirst	0,2525	0,3449	0,3020
	Dir-PosAll	**0,2544**	**0,3533**	**0,3081**
	CTR	0,2513	0,3429	0,2898
WT10G	Dir	0,2094	0,3420	0,3040
	Dir-PosFirst	**0,2164**	0,3794	0,3378
	Dir-PosAll	0,2145	**0,3800**	**0,3391**
	CTR	0,2098	0,3760	0,3270

Table 7. Comparaison entre les différents modèles avec les requêtes longues

Collection	Modèle	MAP	P@10	P@20
AP88	JM	0.3123	0.3959	0.3612
	JM-PosFirst	0.3232	0. 4082	0.3663
	JM-PosAll	**0.3277**	**0.4102**	**0.3673**
	CTR	0.3197	0.3954	0.3633
WT10G	JM	0.1067	0.2280	0.1890
	JM-PosFirst	0.1226	0.2500	0.1990
	JM-PosAll	**0.1257**	**0.2560**	**0.2040**
	CTR	0.1118	0.2384	0.1897

D'après les résultats présentés dans les tables ci-dessus, nous pouvons tirer les remarques et les conclusions suivantes:

- Les deux modèles (le notre et le modèle CTR) améliorent les modèles de base. Cela montre que l'utilisation de la position du terme dans le document dans l'objectif de surpondérer les termes qui apparaissent au début du document peut être utile pour la RI.
- Notre modèle, avec ces deux versions, particulièrement la version PosAll surpasse le modèle CTR sur les deux collections utilisées et cela en termes de MAP, de P@10 et de P@20. Cela montre, d'une part, que l'utilisation de la fonction gaussienne est pertinent pour modéliser le poids d'une position dans le document. D'autre part, la prise en compte de toutes les positions du terme dans le document est plus bénéfique que la prise en compte uniquement de la position de la première occurrence du terme dans le document.

5. Conclusion

Dans cet article, nous avons décrit une extension du modèle de langue pour la RI avec les positions du terme. L'idée principale est de surpondérer les termes qui apparaissent au début du document. Nous avons proposé deux configurations pour mettre en œuvre cette idée. La première considère uniquement la position de la première occurrence du terme dans le document. La seconde prend en compte toutes les positions du terme dans le document. Les deux configurations sont formalisées en tant que modèle de document, qui est ensuite intégré dans un modèle de langue classique. Les deux méthodes de lissage prédominantes: Dirichlet et Jelinek-Mercer, sont utilisées dans ce modèle de langue.

Les résultats obtenus sur deux collections de test TREC ont montré, premièrement, des améliorations significatives par rapport aux deux modèles de langue de base: Dirichlet et Jelinek-Mercer. Cela montre que le facteur position du terme est utile pour la recherche d'informations sous l'hypothèse suivante : "les termes de la requête tendent à apparaitre plus au début des documents pertinents". Deuxièmement, notre modèle, en particulier la version PosAll, surpasse un des modèles de l'état de l'art, qui est le modèle CTR. Cela indique que la prise en compte de toutes les positions du terme dans le document et notre formalisation de ce facteur, avec la fonction gaussien, en tant que modèle de document est plus efficace que la formalisation et l'hypothèse du modèle CTR.

Dans le futur, nous prévoyons d'explorer différents points. Premièrement, l'utilisation d'autres fonctions (linéaire, parabolique, etc.) pour calculer ce nouveau facteur: le poids de la position du terme dans le document. Deuxièmement, l'introduction de ce nouveau facteur dans le processus d'expansion de requêtes.

6. Bibliographie

Acun B., Başpınar A., Oğuz E., Saraç M.İ., Can F. (2013) Topic Tracking Using Chronological Term Ranking. *In: Gelenbe E., Lent R. (eds) Computer and Information Sciences III. Springer*. p. 353-361.

Gerani S., Carman MJ., Crestani F., (2010). Proximity-based Opinion Retrieval". *In ACM International Conference on Research and Development in Information Retrieval,* 2010, Geneva, Switzerland.

Hammache A., Boughanem M., Ahmed-Ouamer R., (2014). Combining compound and single terms under language model framework. *Knowl. Inf. Syst*, Vol 39, n° 2, p. 329-349.

Hulth A., (2003). Improved automatic keyword extraction given more linguistic knowledge. *In Proceedings of conference on Empirical methods in natural language processing*, 2003 Stroudsburg, PA, USA.

Kacem A., Boughanem M., Faiz R., (2017). Emphasizing Temporal-based User Profile Modeling in the Context of Session Search. In ACM Symposium on Applied Computing, 2017, Marrakesh, Morocco.

Kim JY., Croft WB., 2012. A Field Relevance Model for Structured Document Retrieval. *In Proceedings of the 34th European Conference on Advances in Information Retrieval, 2012,* Barcelona, Spain.

Lioma, C., J. G. Simonsen, B. Larsen, N. D. Hansen (2015). Non-compositional term dependence for information retrieval. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2015, New York, NY, USA.

Luhn HP., (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, p. 159–168.

Lv Y., Zhai C., (2009). Positional language models for information retrieval, *In ACM International Conference on Research and Development in Information Retrieval,* 2009, Boston, Massachusetts.

Macdonald C., He B., (2008). Researching and building IR applications using terrier. *Proceeding European conference on Advances in information retrieval*, 2008, Glasgow, UK.

Metzler D., Croft WB., (2005). A Markov random field model for term dependencies. *In ACM International Conference on Research and Development in Information Retrieval,2005,* Salvador, Brazil.

Oglivie P., Callan J., (2003). Combining document representations for known-item search. *In ACM International Conference on Research and Development in Information Retrieval,2003,* Toronto, Canada.

Ponte JM., Croft WB., (1998). A language modeling approach to information retrieval. *In ACM International Conference on Research and Development in Information Retrieval,* 1998, Melbourne, Australia.

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

Robertson SE., Zaragoza, H.,Taylor M., (2004). Simple bm25 extension to multiple weighted fields. In Conference on Information and Knowledge Management, 2004, Washington, DC, USA.

Seo J., Jeon J., (2009). High precision retrieval using relevance-flow graph. *In ACM International Conference on Research and Development in Information Retrieval,* 2009, Boston, Massachusetts.

Song F., Croft WB., (1999). A general language model for information retrieval. *In ACM International Conference on Research and Development in Information Retrieval,* 1999, Berkeley, CA, USA.

Srikanth M., Srihari R., (2002). Biterm language models for document retrieval. *In ACM International Conference on Research and Development in Information Retrieval,* 2002, Tampere, Finland.

Troy AD., Zhang GQ., (2007). Enhancing Relevance Scoring with Chronological Term Rank. *In ACM International Conference on Research and Development in Information Retrieval,* 2007, Amsterdam, Pays-Bas.

Xue XB., Zhou ZH., (2006). Distributional Features for Text Categorization. *In European Conference on Machine Learning,* 2006, Berlin, Germany.

Zamani H., Mitra B., Song X., Craswell N., Tiwary S., (2018). Neural Ranking Models with Multiple Document Fields . *In Proceedings of the 11th ACM International Conference on Web Search and Data Mining, 2018,* Los Angeles, USA.

Zhai C., Lafferty J., (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS),* Vol 22 n° 2, p. 179-214.

Zhao J., Yun Y., (2009). A proximity language model for information retrieval. *In ACM International Conference on Research and Development in Information Retrieval,*2009, Boston, Massachusetts.