Quantification et identification des concepts implicites d'une requête

Romain Deveaud [1] — **Ludovic Bonnefoy** [1] — **Patrice Bellot** [2]

[1] *LIA - Université d'Avignon*
{romain.deveaud,ludovic.bonnefoy}@univ-avignon.fr
[2] *LSIS - Aix-Marseille Université*
patrice.bellot@lsis.org

RÉSUMÉ. *Nous proposons dans cet article une méthode non supervisée pour l'identification et la modélisation de concepts associés à une recherche d'information. Nous utilisons l'allocation de Dirichlet latente (LDA), un modèle génératif probabiliste, pour détecter les concepts implicites de la requête en utilisant les documents obtenus par un processus de retour de pertinence simulé (ou documents de* feedback*). Notre approche estime automatiquement le nombre de concepts ainsi que le nombre de documents de* feedback *sans aucun apprentissage préalable ni paramétrage. Les concepts implicites sont pondérés afin de refléter leur importance relative par rapport à la requête et sont utilisés pour modifier l'ordre des documents renvoyés à l'utilisateur. Nous utilisons quatre sources d'information générales de natures différentes (web, journalistique, encyclopédique) à partir desquelles les documents de* feedback *sont extraits. Nous comparons différentes approches état-de-l'art sur deux collections* ad-hoc *de TREC, et les résultats montrent que l'utilisation de concepts implicites identifiés par notre méthode améliore significativement les performances de recherche documentaire.*

ABSTRACT. *In this paper we introduce an unsupervised method for mining and modeling latent search concepts. We use Latent Dirichlet Allocation (LDA), a generative probabilistic topic model, to exhibit highly-specific query-related topics from pseudo-relevant feedback documents. Our approach automatically estimates the number of latent concepts as well as the needed amount of feedback documents, without any prior training step. Latent concepts are then weighted to reflect their relative adequacy and are further used to automatically reformulate the initial user query. We also explore the use of different types of sources of information for modeling the latent concepts. For this purpose, we use four general sources of information of various nature (web, news, encyclopedic) from which the feedback documents are extracted. We evaluate our approach over two large ad-hoc TREC collections, and results show that it significantly improves document retrieval effectiveness while best results are achieved by combining latent concepts modeled from all available sources.*

MOTS-CLÉS : *Recherche contextuelle, modélisation thématique, retour de pertinence*

KEYWORDS: *Contextual search, topic modeling, relevance feedback*

1. Introduction

Le but de la Recherche d'Information (RI) est de satisfaire le besoin d'information d'un utilisateur, généralement en proposant des documents ou des passages provenant d'une collection cible. Ce besoin est habituellement représenté par une requête composée de quelques mots-clés, qui est soumise au système de recherche d'information. Le système cherche alors les documents qui contiennent les mots-clés, afin de fournir à l'utilisateur une liste de documents ordonnée en fonction de leur pertinence estimée par rapport à la requête. Seulement, un besoin d'information complet peut être trop complexe pour être exprimé en quelques mots, ou l'utilisateur peut ne pas avoir le vocabulaire ou les compétences nécessaires pour formuler efficacement la requête. Ingwersen (1994) dit en effet que la formulation d'une requête par un utilisateur est la représentation de son état cognitif actuel concernant un besoin d'information. Une requête peut ne pas être correctement formulée si l'utilisateur cherche des informations sur une thématique pour laquelle il n'a pas de connaissances. Ainsi, sans contexte additionnel, le système de recherche d'information peut manquer des nuances ou des détails que l'utilisateur n'a pas fourni dans la requête. Ce contexte peut prendre la forme d'un modèle des intérêts de l'utilisateur basé sur son historique personnel (ou ses interactions sociales), ou peut être composé d'éléments extraits de documents similaires représentant les thèmes de la recherche (Finkelstein *et al.,* 2002, White *et al.,* 2009).

Ce deuxième type de contexte est plus généralement connu sous le nom de « recherche d'information conceptuelle » et a reçu beaucoup d'attention au cours de ces dernières années (Bai *et al.,* 2007, Bendersky *et al.,* 2011, Chang *et al.,* 2006, Egozi *et al.,* 2011, Metzler *et al.,* 2007). L'idée générale est d'étendre les requêtes avec des ensembles de mots ou de multi-mots extraits de documents de *feedback*. L'ensemble de *feedback* est composé de documents qui sont pertinents ou pseudo-pertinents par rapport à la requête initiale, et qui sont à même de contenir des informations importantes sur le contexte de la recherche. Les mots exprimant le plus d'information par rapport à la requête sont traités comme des concepts implicites. Ils sont alors utilisés pour reformuler la requête. Le problème avec cette approche est que chaque mot représente un concept spécifique. Seulement un concept représente une notion et peut être vu comme un ensemble de connaissances. Stock (2010) donne une définition qui suit cette direction en affirmant qu'un concept est défini comme une classe contenant des objets possédant certaines propriétés et attributs.

L'objectif du travail présenté dans cet article est de représenter avec précision les concepts sous-jacents associés à une requête, améliorant indirectement les informations contextuelles liées à la recherche documentaire. Nous introduisons ainsi une méthode entièrement non supervisée qui permet de détecter les concepts implicites liés à une requête donnée et d'améliorer les performances d'un système de recherche documentaire en incorporant ces concepts à la requête initiale. Pour chaque requête, les concepts implicites sont extraits d'un ensemble réduit de documents de *feedback* initialement récupérés par le système. Ces documents de *feedback* peuvent venir de la collection cible ou de n'importe quelle autre source d'information textuelle. Elle

ne requiert aucun paramétrage préalable, et quantifie automatiquement le nombre de concepts ainsi que le nombre de documents de *feedback* nécessaires.

birds		**comic**		**toys**		**paleontology**	
$P(w\|k)$	word w	$P(w\|k)$	word w	$P(w\|k)$	word w	$P(w\|k)$	word w
0.196	feathers	0.257	dinosaur	0.370	dinosaur	0.175	dinosaur
0.130	birds	0.180	devil	0.165	price	0.125	kenya
0.112	evolved	0.095	moon-boy	0.112	party	0.122	years
0.102	flight	0.054	bakker	0.053	birthday	0.087	fossils
0.093	dinosaurs	0.054	world	0.039	game	0.082	paleontology
0.084	protopteryx	0.049	series	0.023	toys	0.072	expedition
0.065	fossil	0.045	marvel	0.021	t-rex	0.070	discovery
	...		...		...		...
$\hat{\delta}_0 = 0.434$		$\hat{\delta}_1 = 0.254$		$\hat{\delta}_2 = 0.021$		$\hat{\delta}_3 = 0.291$	

Tableau 1. *Concepts identifiés pour la requête « dinosaurs » (topic 14 de la Web Track de TREC) par l'approche présentée dans ce papier. Les mots sont pondérés pour refléter leur informativité au sein d'un même concept k. Les concepts sont également pondérés selon leur cohérence par rapport à la requête. Les étiquettes ont été définies manuellement par souci de clarté.*

L'exemple présenté dans le Tableau 1 montre les concepts implicites identifiés par notre approche pour la requête « *dinosaurs* » en utilisant une large collection de documents web comme source d'information. Chaque concept k est composé de mots w qui sont liés thématiquement et pondérés par leur probabilité $P(w|k)$ d'appartenance à ce concept. Cette pondération accentue les mots importants et permet de refléter efficacement leur influence au sein du concept. L'extraction de concept est effectuée en utilisant l'allocation de Dirichlet latente (LDA) (Blei *et al.,* 2003), un modèle génératif probabiliste. Étant donné une collection de documents, LDA calcule les distributions des concepts au sein des documents et les distributions des mots au sein des concepts. Nous pondérons les concepts eux-mêmes afin de refléter leur cohérence au sein de l'ensemble de documents de *feedback*. Un poids inférieur est assigné aux concepts de moindre importance qui apparaissent dans des documents ayant une faible probabilité d'apparition par rapport à la requête. Dans notre exemple, le concept « toys » parait peu important pour préciser le contexte d'une requête traitant de dinosaures. Son poids ($\hat{\delta}_2 = 0,021$) reflète donc la faible probabilité que ce concept soit celui qui concerne la requête. Néanmoins, le système sera tout de même capable de récupérer des documents pertinents dans le cas où l'utilisateur chercherait vraiment des jouets de dinosaures.

L'avantage principal de notre approche est qu'elle est entièrement non supervisée et qu'elle ne requiert aucun entraînement. Le nombre de documents de *feedback* nécessaires ainsi que le nombre de concepts sont automatiquement estimés au moment où la requête est soumise au système. Nous insistons sur le fait que les algorithmes ne disposent d'aucune information préalable au sujet de ces concepts. Aucun travail d'annotation n'a été réalisé sur les requêtes et à aucun moment nous ne fixons manuellement des paramètres, à l'exception du nombre de mots composant les concepts.

La suite de cet article est organisée comme suit. Nous détaillons quelques travaux connexes de modélisation thématique appliquée à la RI dans la section 2. La section 3 présente rapidement l'allocation de Dirichlet latente, puis détaille l'approche que nous proposons. La section 4.1 donne un aperçu des sources d'information générales que nous utilisons pour modéliser les concepts implicites. Nous évaluons notre approche et discutons les résultats dans la section 4. Pour finir, la section 5 conclut cet article et offre quelques perspectives.

2. Travaux connexes

Le travail présenté dans cet article est une approche originale de modélisation thématique qui utilise des informations provenant de sources textuelles diverses et ayant pour but d'améliorer la qualité de la recherche documentaire. La modélisation thématique probabiliste (et notamment l'allocation de Dirichlet latente) pour la RI a été largement utilisée récemment de diverses manières (Andrzejewski *et al.,* 2011, Lu *et al.,* 2011, Park *et al.,* 2009, Wei *et al.,* 2006, Yi *et al.,* 2009), et toutes les études rapportent des amélioration des performances de recherche documentaire. L'idée principale est de classifier *a priori* la collection de documents dans sa totalité, puis ensuite d'identifier les thèmes (ou concepts) liés à la requête. Le modèle de langue de chaque document est alors lissé en incorporant les probabilités d'appartenance des mots à ces thèmes (Lu *et al.,* 2011, Wei *et al.,* 2006, Yi *et al.,* 2009). D'autres approches essayent quant à elles d'enrichir directement la requête avec les mots qui appartiennent à ces concepts pseudo-pertinents (Andrzejewski *et al.,* 2011, Park *et al.,* 2009). L'idée d'utiliser des documents de *feedback* a été explorée par (Andrzejewski *et al.,* 2011), où des concepts spécifiques à la requête sont extraits des deux premiers documents renvoyés par la requête originale. Ces concepts sont identifiés en utilisant les distributions précédemment calculées par LDA sur la collection entière. La requête est finalement enrichie avec les mots appartenant aux concepts apparaissant dans les deux premiers documents de *feedback*. Seulement, toutes ces approches nécessitent de fixer de nombreux paramètres, comme le nombre de concepts ou encore le nombre de documents utilisés. Nous nous plaçons dans un contexte évolutif et proposons des alternatives pour estimer automatiquement ces paramètres. À notre connaissance, notre approche est également la première tentative de modélisation thématique effectuée uniquement sur des documents de *feedback* et au moment de la requête, contrairement aux approches traditionnelles qui utilisent une collection entière.

3. Quantification et identification de concepts implicites

Nous proposons de modéliser les concepts implicites à un besoin d'information et de les utiliser pour améliorer la représentation de la requête. Soit $\mathcal{R}$ une source d'information textuelle à partir de laquelle les concepts implicites vont être extraits. Un sous-ensemble initial $\mathcal{R}_Q$ est formé par les documents de *feedback* les mieux classés par rapport à une requête Q lors d'une première étape de recherche. L'algorithme de RI peut être de n'importe quel type, le point important est que $\mathcal{R}_Q$ est une collection réduite qui ne contient qu'un petit nombre de documents traitant de thématiques communes.

L'allocation de Dirichlet latente (Blei *et al.,* 2003) (LDA) est un algorithme de modélisation thématique probabiliste qui considère les documents comme des ensembles de concepts, et les concepts comme des ensembles de mots. Utiliser LDA sur un ensemble de documents extraits grâce à la requête offre l'avantage de modéliser les concepts qui lui sont très fortement liés. De nombreux problèmes doivent être résolus afin de modéliser ces concepts en vue de leur utilisation pour rechercher des documents. Premièrement, comment estimer le bon nombre de concepts ? LDA est un algorithme non supervisé mais nécessite quelques paramètres comme le nombre de concepts. Seulement, le nombre de concepts apparaissant dans un ensemble de documents de *feedback* est dépendant de la collection et surtout de la requête. Nous avons donc besoin d'estimer le nombre de concepts implicites de chaque requête. De même, quelle quantité de documents de *feedback* doit être choisie pour s'assurer que les concepts extraits sont effectivement liés à la requête ? En d'autre mots : comment idéalement éviter les concepts bruités et non pertinents ? Troisièmement, les différents concepts n'ont pas la même influence par rapport à un besoin d'information. Le même problème apparaît au sein des concepts où certain mots sont plus importants que d'autres. La pondération des mots et des concepts est ainsi essentielle pour refléter leur importance contextuelle. Enfin, comment utiliser ces concepts implicites pour améliorer la recherche de documents ? Comment peuvent-ils s'intégrer à un algorithme de RI existant ?

Nous décrivons notre approche et répondons à ces questions dans cette section. Une évaluation détaillée ainsi qu'une analyse des différents paramètres estimés sont proposées dans la section 4.

3.1. *Allocation de Dirichlet latente*

L'allocation de Dirichlet latente est un modèle thématique génératif probabiliste (Blei *et al.,* 2003). Il se base sur l'intuition que les documents sont composés de plusieurs thèmes (et non pas de mots), où un thème est une distribution multinomiale sur un vocabulaire fixé W. Le but de LDA est ainsi de découvrir les thèmes présents au sein d'une collection de documents. Les documents de la collection sont modélisés comme des ensembles de K thèmes qui sont eux-même des distributions multinomiales sur W. La distribution thématique ϕ_k d'un thème k est générée par une loi de Dirichlet avec un paramètre β, tandis que la distribution θ_d d'un document d est générée par une loi de Dirichlet avec un paramètre α. En d'autres mots, $\theta_{d,k}$ est la probabilité que le thème k apparaisse dans le document d (i.e. $P(k|d)$). Respectivement, $\phi_{k,w}$ est la probabilité que le mot w appartienne au thème k (i.e. $P(w|k)$). Différentes méthodes d'approximation ont été développées (Blei *et al.,* 2003, Griffiths *et al.,* 2004) ; nous utilisons dans ce travail l'algorithme implémenté et distribué par le Pr. Blei[1].

1. `http://www.cs.princeton.edu/~blei/lda-c`

3.2. *Estimer le nombre de concepts*

Différents concepts implicites peuvent représenter un besoin d'information, et leur nombre dépend de la richesse ou de l'ambiguïté de ce besoin. LDA permet de modéliser la distribution thématique d'une collection donnée, mais le nombre de concepts est un paramètre qui doit être fixé. Seulement on ne peut savoir à l'avance le nombre de concepts liés à une requête. Nous proposons une méthode qui estime automatiquement le nombre de concepts implicites.

En partant du principe que les concepts identifiés par LDA sont représentés par les n mots qui ont les plus fortes probabilités, nous définissons un opérateur $\mathrm{argmax}[n]$ qui produit les n arguments obtenant les plus fortes valeurs pour une fonction donnée. Nous pouvons ainsi obtenir l'ensemble $\mathbb{W}_k$ des n mots qui ont les plus fortes probabilités $P(w|k) = \phi_{k,w}$ dans le concept k :

$$\mathbb{W}_k = \underset{w}{\mathrm{argmax}}[n]\ \phi_{k,w}$$

Différents travaux ont été réalisés pour trouver le bon nombre de concepts contenus dans une collection de documents (Arun *et al.,* 2010, Cao *et al.,* 2009). Même si ils diffèrent sur certains points, ils suivent tous un même principe qui revient à calculer des similarités (ou des distances) entre toutes les paires de concepts pour différents modèles obtenus en faisant varier le nombre de concepts. Ainsi, pour le même ensemble de documents de *feedback* $\mathcal{R}_Q$, différents modèles LDA sont calculés en faisant varier le nombre de concepts de 1 à 20. Pour chacun de ces modèles, nous calculons alors la somme des divergences $D(k_i||k_j)$ entre tous les paires de concepts (k_i, k_j) afin de déterminer à quels points les concepts sont correctement délimités. Finalement, nous ne choisissons que le modèle pour lequel la divergence globale est la plus forte, car c'est celui qui propose la meilleure démarcation entre les concepts. Le nombre de concepts $\hat{K}$ estimé par notre méthode est donné par la formule suivante :

$$\hat{K} = \underset{K}{\mathrm{argmax}}\ \frac{1}{K(K-1)} \sum_{(k_i,k_j)\in\mathbb{T}_K} D(k_i||k_j)$$

où K est le nombre de concepts donné en paramètre à LDA, et $\mathbb{T}_K$ est l'ensemble des K concepts. Ainsi, $\hat{K}$ est le nombre de concepts qui permet d'obtenir la meilleure démarcation entre les concepts pour l'ensemble de documents $\mathcal{R}_Q$: c'est le nombre de concepts implicites de la requête Q formulée par l'utilisateur. La divergence de Kullback-Leibler mesure la dissimilarité entre deux distributions de probabilités. Elle est utilisée en particulier par LDA afin de minimiser la variation thématique entre deux itérations de l'algorithme d'espérance-maximisation (Blei *et al.,* 2003), ainsi que dans d'autres domaines pour mesurer des similarités entre des distributions de mots (AlSumait *et al.,* 2008). Nous utilisons la version symétrique de la divergence de Kullback-Leibler afin d'éviter des problèmes évidents lors du calcul de divergences entre toutes les paires de concepts :

$$D(k_i||k_j) = \sum_{w\in\mathbb{W}_{inter}} P(w|k_i)\log\frac{P(w|k_i)}{P(w|k_j)} + \sum_{w\in\mathbb{W}_{inter}} P(w|k_j)\log\frac{P(w|k_j)}{P(w|k_i)}$$

où $\mathbb{W}_{inter} = \mathbb{W}_{k_i} \cap \mathbb{W}_{k_j}$. Les probabilités des mots sachant les concepts sont obtenues à partir de la distribution multinomiale ϕ_k. La sortie finale est le nombre estimé de concepts implicites $\hat{K}$ ainsi que l'ensemble de concepts $\mathbb{T}_{\hat{K}}$ qui lui est associé. Nous définissons cet ensemble de concepts comme étant un *modèle conceptuel*.

3.3. *Combien de documents de feedback ?*

Un problème récurent avec les approches à base de retour de pertinence simulé est que des documents non pertinents peuvent être inclus dans les documents de *feedback*. Ce problème est d'autant plus important dans le cadre de notre approche puisqu'il pourrait conduire à la modélisation de concepts qui ne sont pas liés à la requête initiale. Nous réduisons l'impact de ce problème principalement en réduisant le nombre de documents de *feedback*. En effet, les documents pertinents ont généralement une concentration plus élevée dans les premiers rangs de la liste. Ainsi une manière simple de réduire les chances d'avoir des documents de *feedback* non pertinents est de réduire leur nombre.

Seulement, un même nombre ne peut pas être choisi arbitrairement pour toutes les requêtes. Certains besoins d'information peuvent être satisfaits par 2 ou 3 documents, tandis que d'autres peuvent en requérir 15 ou 20. Le choix du nombre de documents de *feedback* doit donc être automatique pour chaque requête. Dans ce but, nous comparons les modèles conceptuels générés à partir de différents nombres m de documents de *feedback*. Afin d'éviter le bruit et les concepts non pertinents, nous favorisons les modèles conceptuels qui contiennent des concepts similaires à ceux présents dans les autres modèles. Notre hypothèse est que tous les documents de *feedback* discutent de concepts similaires ou liés, peu importe le nombre de documents. Ainsi, des concepts apparaissant dans différents modèles appris sur différents ensembles de documents de *feedback* sont certainement liés à la requête, tandis que des concepts bruités ont peu de chances d'apparaître à chaque fois.

Nous estimons la similarité entre deux modèles conceptuels en calculant les similarités entre toutes les paires de concepts des deux modèles. Seulement, deux modèles différents sont générés à partir de documents différents, ils ne partagent donc pas le même espace probabiliste. Les distributions de probabilités ne sont donc pas comparables, le calcul de similarité ne peut se faire qu'en prenant en compte les mots des concepts. Les concepts sont donc ramenés à de simples sacs de mots, et nous utilisons une mesure de similarité basée sur la fréquence inverse des mots dans les documents de la collection :

$$sim(\mathbb{T}_{\hat{K}(m)}, \mathbb{T}_{\hat{K}(n)}) = \sum_{k \in \mathbb{T}_{\hat{K}(m)}} \sum_{k' \in \mathbb{T}_{\hat{K}(n)}} \frac{|k \cap k'|}{|k|} \sum_{w \in W} \log \frac{N}{df_w}$$

où $\frac{|k \cap k'|}{|k|}$ est le recouvrement en mots entre les deux concepts, df_w est la fréquence documentaire du mot w dans la collection cible, et N est le nombre total de documents dans la collection. Le but initial de cette mesure était la détection de la nouveauté (i.e. minimisation de la similarité) entre deux phrases (Metzler *et al.,* 2005a), ce qui est

précisément ce que nous cherchons, à l'exception près que nous voulons détecter la redondance (i.e. maximiser la similarité).

La somme finale des similarités entre chaque paire de concepts produit le score de similarité du modèle conceptuel courant par rapport à tous les autres. Le modèle conceptuel qui maximise cette similarité est considéré comme le meilleur candidat pour représenter les concepts implicites d'une requête. Autrement dit, les M premiers documents de *feedback* sont utilisés pour modéliser les concepts, où :

$$M = \underset{m}{\operatorname{argmax}} \sum_{n} sim(\mathbb{T}_{\hat{K}(m)}, \mathbb{T}_{\hat{K}(n)})$$

Ainsi, pour chaque requête, le modèle conceptuel qui est le plus similaire à tous les autres modèles devient l'ensemble de concepts implicites liés à la requête utilisateur.

Cette méthode fait appel de nombreuses fois à l'algorithme LDA et l'on pourrait se poser la question du temps de calcul et de la pertinence d'une telle approche. Traditionnellement, calculer un modèle LDA sur une collection de plusieurs millions de documents peut prendre plusieurs heures. Concernant notre approche, les modèles conceptuels sont appris sur un très petit nombre de documents (typiquement entre 1 et 20) et ne sont donc pas sensibles aux problèmes de complexité algorithmique.

3.4. *Pondération des concepts*

Différents concepts peuvent être liés à une requête utilisateur, mais tous n'ont pas la même importance. Par exemple, notre méthode se base sur des estimations et peut donc potentiellement modéliser des concepts peu pertinents ou bruités. Il est donc essentiel de promouvoir les concepts appropriés et de déprécier ceux qui ne le sont pas. Nous classons ainsi les concepts par ordre d'importance et nous leur attribuons des poids en conséquence. Nous définissons le score δ_k d'un concept k par :

$$\delta_k = \sum_{D \in \mathcal{R}_Q} P(Q|D)P(k|D)$$

où $\mathcal{R}_Q$ est l'ensemble des documents de *feedback* choisis dans la section précédente. La probabilité $P(k|D)$ que le concept k apparaisse dans le document D est donnée par la distribution multinomiale θ apprise précédemment par LDA.

Après avoir pondéré les concepts, nous améliorons cette représentation en pondérant les mots qui les composent. En effet ces mots n'ont pas tous la même importance relative au sein d'un même concept. Nous utilisons logiquement la distribution multinomiale ϕ_k apprise par LDA qui donne la probabilité d'appartenance de chaque mot du vocabulaire au concept k. Après normalisation, le poids du mot w dans le concept k est donné par :

$$\hat{\phi}_{k,w} = \frac{P(w|k)}{\sum_{w' \in \mathbb{W}_k} P(w'|k)}$$

où $\mathbb{W}_k$ est l'ensemble des mots du concept k tel que défini dans la section 3.2. Au final, un concept appris par notre approche est en réalité un ensemble de mots pondérés

représentant un aspect du besoin d'information sous-jacent à la requête utilisateur. Le concept est lui-même pondéré afin de refléter son importance relative par rapport aux autre concepts.

3.5. *Intégration des concepts pour la recherche documentaire*

Il y a de nombreuses façons de prendre en compte des aspects conceptuels pour la recherche documentaire. Ici, le score final d'un document D par rapport à une requête utilisateur Q est déterminé par la combinaison linéaire d'une recherche standard des mots de la requête et d'une recherche des concepts implicites :

$$s(Q,D) = \lambda \cdot P(Q|D) + (1-\lambda) \cdot \prod_{k \in \mathbb{T}_{\hat{K}(M)}} \hat{\delta}_k \prod_{w \in \mathbb{W}_k} \hat{\phi}_{k,w} \cdot P(w|D) \qquad [1]$$

où $\mathbb{T}_{\hat{K}(M)}$ est le *modèle conceptuel* qui comprend les concepts implicites de la requête Q et $\hat{\delta}_k$ est le poids normalisé du concept k :

$$\hat{\delta}_k = \frac{\delta_k}{\sum_{k' \in \mathbb{T}_{\hat{K}}} \delta_{k'}}$$

Nous utilisons dans ce travail une approche par modèles de langue pour la RI (Lavrenko *et al.,* 2001). $P(w|D)$ est ainsi l'estimation par maximum de vraisemblance du mot w dans le document D, calculée en utilisant le modèle de langue du document D dans la collection cible $\mathcal{C}$. Le modèle standard de recherche par modèle de langue $P(Q|D)$ peut être décomposé comme $P(Q|D) = \prod_{w \in Q} P(w|D)$. Le problème des probabilités nulles est réglé grâce au lissage standard de Dirichlet qui se trouve être plus efficace pour les requêtes composées de mots-clés (contrairement aux requêtes formulées en langage naturel) (Zhai *et al.,* 2004), ce qui est le cas dans cette étude. Nous fixons le paramètre du lissage à 1500 et nous ne le faisons pas varier au cours de nos expérimentations. Bien que nous ayons choisi une approche par modèle de langue, il est important de noter que ce modèle est générique et que la méthode d'appariement pourrait être entièrement substituée à une autre méthode état-de-l'art (comme BM25 (Robertson *et al.,* 1994) ou les modèles fondés sur l'information (Clinchant *et al.,* 2010)) sans changer la façon dont les concepts implicites agissent sur le classement des documents.

4. Évaluation

Nous présentons dans cette section les différentes expériences que nous avons menées dans le cadre de ce travail. Nous commençons par détailler les différentes sources d'information utilisées pour identifier les concepts dans la section 4.1. Nous expliquons ensuite notre protocole expérimental ainsi que les collections de test que nous utilisons pour l'évaluation.

4.1. *Sources d'information pour l'identification de concepts*

L'approche présentée dans cet article nécessite une source d'information à partir de laquelle les concepts peuvent être extraits. Cette source peut être la collection cible, comme dans les approches traditionnelles de retour de pertinence, ou une collection externe. Dans ce travail nous utilisons plusieurs sources différentes et suffisamment importantes pour traiter d'un très large spectre de concepts. Ainsi nous pouvons explorer quels effets ont la nature, la taille ou la qualité de chaque source sur l'identification des concepts.

Notre ensemble de sources d'information est composé de quatre ressources générales : Wikipédia comme source encyclopédique, le corpus LDC du New York Times et le corpus LDC du GigaWord comme sources d'articles de presse, et la catégorie B du ClueWeb09[2] comme source de pages web. Le corpus GigaWord LDC (version anglaise) est constitué de 4 111 240 dépêches collectées dans quatre sources internationales distinctes (Graff *et al.,* 2003). Le corpus LDC du New York Times contient 1 855 658 articles de presse publiés entre 1987 et 2007 (Sandhaus, 2008). La collection Wikipédia est une capture de l'encyclopédie en ligne datant du mois de juillet 2011 qui contient 3 214 014 documents[3]. Enfin, nous avons supprimé les documents considérés comme « spam » de la catégorie B du ClueWeb09 en nous basant sur une liste standard éditée pour cette collection[4]. Nous avons suivi pour cela les recommandations des auteurs (Cormack *et al.,* 2011) et avons fixé le paramètre contrôlant l'influence du spam à 70. Le corpus résultant de cette opération est composé de 29 038 220 pages web.

Ressource	# documents	# mots unique	# total de mots
NYT	1 855 658	1 086 233	1 378 897 246
Wiki	3 214 014	7 022 226	1 033 787 926
GW	4 111 240	1 288 389	1 397 727 483
Web	29 038 220	33 314 740	22 814 465 842

Tableau 2. *Récapitulatif des quatre sources d'information générales utilisées.*

4.2. *Cadre expérimental*

Nous avons évalué notre approche en utilisant deux collections utilisées dans des tâches majeures de TREC[5] ; elles sont détaillées dans le tableau 3. La collection Robust04 est composée d'articles de presse provenant de divers journaux et a été utilisée dans le cadre de la tâche Robust en 2004. Elle comprend notamment les corpus suivants : FT (Financial Times), FR (Federal Register 94), LA (Los Angeles Times) et FBIS (i.e. les disques TREC 4 et 5, sans la partie concernant le Congressional Record). Elle contient également 250 *topics* et des relevés de pertinence complets. Quant

2. http://boston.lti.cs.cmu.edu/clueweb09/
3. http://dumps.wikimedia.org/enwiki/20110722/
4. http://plg.uwaterloo.ca/~gvcormac/clueweb09spam/
5. http://trec.nist.gov

au ClueWeb 09, c'est la plus grande collection de pages web mise à disposition de la communauté de RI à l'heure où nous écrivons cet article. Cette collection a été utilisée dans plusieurs tâches de TREC comme par exemple les tâches Web, Blog et Million Query. Nous considérons ici uniquement la catégorie B du ClueWeb09 (ClueWeb09-B), qui est composée d'environ 50 millions de pages web. Nous utilisons également l'intégralité des *topics* et des jugements de pertinence de la tâche Web de TREC (années 2009 à 2011) pour notre évaluation.

Nom	**# documents**	**Topics utilisés**
Robust04	528 155	301-450, 601-700
ClueWeb09-B	50 220 423	1-150

Tableau 3. *Résumé des collections de test de TREC utilisées pour notre évaluation.*

Nous avons utilisé Indri[6] pour indexer ces collections et chercher les documents. Les mêmes paramètres ont été utilisés dans tous les cas : les mots ont été légèrement racinisés par l'algorithme standard de Krovetz, et les mots outils présents dans la liste fournie avec Indri ont été supprimés. Comme nous l'avons vu en section 3, les concepts sont composés d'un nombre fixe de mots. Dans ce travail, nous fixons le nombre de mots appartenant à un concept à $n = 10$. Nous utilisons trois mesures d'évaluation standard comme moyen de comparaison : le gain cumulatif réduit normalisé (nDCG@20) et la précision (P@20) aux premiers rangs, ainsi que la précision moyenne de la liste de résultats entière (MAP).

4.3. *Analyse des nombres de concepts et de documents estimés*

La figure 1 présente des histogrammes traçant le nombre de requêtes en fonction du nombre de concepts implicites estimé et du nombre de documents de *feedback*, et ce pour les deux collections. On voit que le comportement est relativement identique sur les deux collections. Entre deux et trois concepts sont identifiés pour la grande majorité des requêtes. De même ces concepts sont généralement identifiés au sein d'un nombre assez réduit de documents, entre deux et quatre pour les deux collections. Il est toutefois intéressant de noter la différence entre le nombre de documents de *feedback* utilisés par les ressources Web et Wikipédia. On peut voir en effet que 2 ou 3 articles Wikipédia suffisent pour un très grand nombre de requêtes, alors qu'un plus grand nombre est nécessaire pour la ressource Web. Ce comportement est très cohérent avec la nature même de Wikipédia, où les articles sont rédigés dans le but d'être très précis et de ne pas trop s'éparpiller. Il est d'ailleurs fréquent qu'un article devenu trop conséquent soit coupé en plusieurs autres articles traitant chacun un sujet très spécifique. Ceci est confirmé par le fait que le nombre de concepts $\hat{K}$ et le nombre de documents M sont fortement corrélés pour Wikipédia selon le test de Pearson : $\rho = 0,7$ pour les requêtes du ClueWeb09 et $\rho = 0,616$ pour Robust04 (avec une valeur

6. `http://www.lemurproject.org`

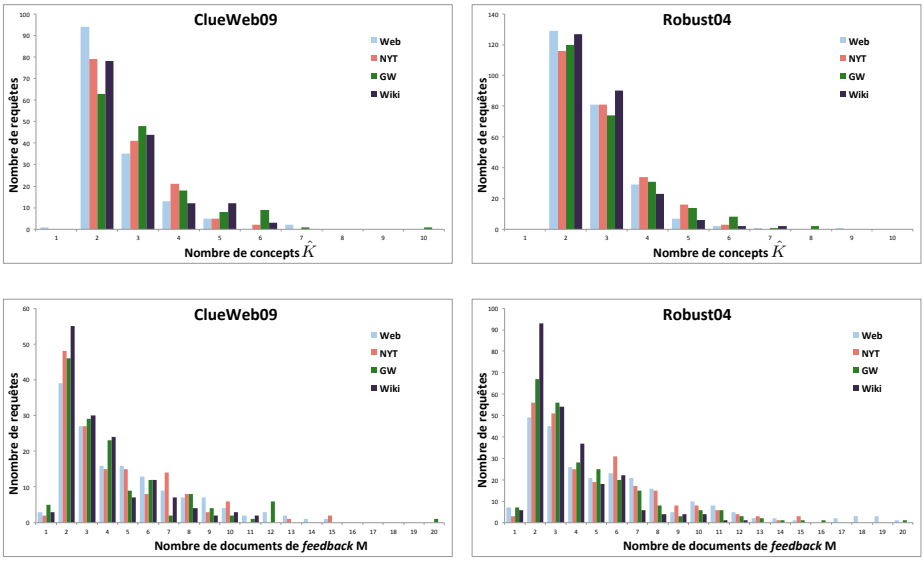

Figure 1. *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{K}$ de concepts implicites (section 3.2) et du nombre M de documents de feedback (section 3.3).*

$p < 0,01$ obtenue par un test de permutations). De même, la nature très hétérogène du Web pousse notre méthode à devoir choisir un plus grand nombre de documents de *feedback* afin de pouvoir modéliser correctement les différents concepts implicites. La corrélation est aussi présente pour cette ressource mais elle est moins importante ($\rho = 0,33$ pour le ClueWeb09 et $\rho = 0,39$ pour Robust 04), ce qui reflète cette hétérogénéité et la difficulté à estimer les deux paramètres.

4.4. *Recherche conceptuelle de documents*

Les résultats de recherche documentaire pour les deux collections de test sont présentés dans le tableau 4. Les concepts sont identifiés suivant l'approche présentée dans ce papier et ont un poids égal à celui de la requête initiale ($\lambda = 0.5$ dans l'équation 1, ce qui revient à ne fixer aucun poids). Nous présentons les résultats obtenus en utilisant chaque ressource séparément pour l'identification des concepts. Ces approches sont comparées à deux systèmes de base performants. Le premier est une version mise à jour des modèles de pertinence pour la recherche par modèles de langue (Relevance Models, RM (Lavrenko *et al.,* 2001)) obtenant des résultats état-de-l'art sur différentes collections de test. Ce modèle est performant et enrichit la requête initiale avec les termes les plus informatifs extraits des documents de *feedback* les mieux classés (Deveaud *et al.,* 2012). Le second est le modèle de dépendance séquentielle, un cas particuliers des champs de Markov aléatoires (MRF) pour la RI (Metzler *et al.,* 2005b) qui modélise l'apparition de termes adjacents au sein de la requête. Nous suivons les recommandations des auteurs et fixons les poids à 0,85, 0,10 et 0,05 pour la recherche

d'unigrammes, de bigrammes et de bigrammes à trous respectivement. Ce modèle est reconnu pour avoir obtenu des performances état-de-l'art sur différentes collections de test, dont celles que nous utilisons ici (Clarke *et al.,* 2010, Metzler *et al.,* 2007).

	ClueWeb09-B			**Robust04**		
	nDCG@20	P@20	MAP	nDCG@20	P@20	MAP
MRF	0,2128	0,2838	0,1401	0,4231	0,3612	0,2564
RM	0,2368	0,3095	0,1413	0,4251	0,3725*	0,2764***
GW	0,2098	0,2782	0,1283	0,4521$^{***}_{rrr}$	0,3841$^{**}_{rr}$	0,2820***
Wiki	0,2142	0,2980	0,1408	0,4189	0,3549	0,2632
NYT	0,2144	0,2816	0,1346	**0,4589**$^{***}_{rrr}$	**0,3928**$^{***}_{rrr}$	**0,2891**$^{***}_{rr}$
Web	**0,2529*****	**0,3328*****	**0,1474**	0,4428$^{*}_{r}$	0,3754*	0,2760***
Comb	0,2465***	0,3247***	0,1597$^{***}_{rrr}$	0,4680$^{***}_{rrr}$	0,3969$^{***}_{rrr}$	0,2929$^{***}_{rrr}$

Tableau 4. *Performances de recherche documentaire sur deux collections de test majeures de TREC. Nous utilisons le test apparié de Student (t-test) pour déterminer les différences significatives avec les systèmes de base MRF ($^{*}: p < 0,1$; $^{**}: p < 0,05$; $^{***}: p < 0,01$) et RM ($_{r}: p < 0,1$; $_{rr}: p < 0,05$; $_{rrr}: p < 0,01$). Les lignes GW, Wiki, NYT, Web et Comb correspondent à notre méthode.*

Nous observons dans le tableau 4 que les résultats varient beaucoup en fonction de la ressource utilisée pour identifier les concepts. Pour la recherche web (avec la collection ClueWeb09), le GigaWord, le New York Times et Wikipédia ne permettent pas d'identifier des concepts de qualité. Les meilleurs résultats parmi ces trois-ci sont obtenus soit avec le New York Times, soit avec Wikipédia, et les résultats sont plus ou moins similaires à ceux obtenus par le système MRF. L'utilisation du New York Times et de Wikipédia permet d'obtenir des résultats du niveau du système MRF, tandis le GigaWord se place derrière. Cette contre-performance semble principalement due à sa nature : les dépêches sont très courtes et vont directement aux faits, le vocabulaire n'est pas assez riche pour arriver à modéliser des concepts cohérents. De son côté, la ressource Web obtient de meilleurs résultats très significatifs par rapport aux deux systèmes de base, sauf pour la précision moyenne. Pour la recherche d'articles de presse (avec la collection Robust04), l'influence des quatre ressources est clairement différente. Nous observons que les meilleurs résultats sont obtenus en utilisant les concepts identifiés à partir du NYT et du GigaWord. Le Web obtient également de bons résultats, mais ils sont très inférieurs et moins significatifs que ceux obtenus par NYT et GW.

La nature de la ressource à partir de laquelle les concepts sont identifiés semble ainsi fortement corrélée avec la collection de documents. En effet, la ressource Web permet d'obtenir de meilleurs concepts pour la recherche web tandis que les autres échouent. De la même façon, les ressources journalistiques donnent de meilleurs résultats pour la recherche d'articles de presse. On pourrait penser que les résultats similaires obtenus en utilisant le NYT et le GigaWord sont dus au fait que leur vocabulaire est très similaire, mais elle ne partagent que 18,7% de leurs mots uniques. Bien que les vocabulaires soient très différents, les mêmes concepts peuvent tout de même être identifiés. Tandis que la ressource Web joue un rôle majeur pour les deux types de

recherche documentaire, l'utilisation de Wikipédia échoue dans tous les cas. Nous pensons que ceci est principalement dû au fait que nous n'avons pas de collection de documents de type encyclopédique à chercher. Au vu des autres résultats, nous pensons en effet que l'utilisation de Wikipédia serait efficace pour identifier des concepts dans un contexte de recherche encyclopédique, ce que nous vérifierons dans des travaux futurs.

Nous explorons également les effets d'une combinaison des concepts implicites provenant des quatre ressources en même temps. Nous adaptons la fonction de classement des documents (équation 1) pour qu'elle puisse prendre en compte un ensemble fini de ressources :

$$s(Q,D) = \lambda \cdot P(Q|D) + (1-\lambda) \cdot \frac{1}{|\mathcal{S}|} \prod_{\sigma \in \mathcal{S}} \prod_{k \in \mathbb{T}^{\sigma}_{\hat{K}(M)}} \hat{\delta}_k \prod_{w \in \mathbb{W}_k} \hat{\phi}_{k,w} \cdot P(w|D) \quad [2]$$

où $\mathbb{T}^{\sigma}_{\hat{K}(M)}$ est le *modèle conceptuel* construit par notre approche à partir de la source d'information σ appartenant à l'ensemble $\mathcal{S}$. Notre approche permet de combiner naturellement les ressources, toujours sans aucune étape d'apprentissage.

Les résultats présentés dans le tableau 4 à la ligne **Comb** ne sont pas surprenants et supportent les principes de polyreprésentation (Ingwersen, 1994) et de redondance intentionnelle (Jones, 1990). Ceux-ci affirment que combiner des représentations structurellement et cognitivement différentes du besoin d'information permet d'améliorer les chances de trouver des documents pertinents. Même si la combinaison n'améliore pas les résultats pour toutes les mesures d'évaluation par rapport à la ressource Web, elle atteint tout de même toujours les plus hauts niveaux de significativité par rapport aux deux systèmes de base. Les ressources "mineures" obtiennent des performances limitées quand elles sont utilisées seules, mais elles jouent un rôle essentiel dans la combinaison. Elles apportent chacune des concepts uniques et cohérents qui, ensemble, forment un modèle conceptuel multiple et complet qui couvre les différents aspects de la requête.

5. Conclusion

Nous avons présenté dans cet article une approche entièrement non supervisée pour la quantification et l'identification des concepts implicites d'une requête. Ces concepts sont extraits à partir d'ensembles de documents pseudo-pertinents provenant de plusieurs sources d'information hétérogènes. Le nombre de concepts implicites et le nombre de documents de *feedback* approprié sont automatiquement estimés au moment de l'exécution de la requête, sans apprentissage supervisé ni étiquetage préalable. Globalement, notre méthode obtient de bons résultats, significativement supérieurs aux deux systèmes de base, lorsque les sources d'information sont du même type que la collection de documents. Les meilleurs résultats sont obtenus en combinant les concepts implicites identifiés à partir de toutes les sources disponibles, ce qui montre que notre approche est suffisamment robuste pour traiter des documents hétérogènes abordant différentes thématiques. Notre méthode présente néanmoins de nombreuses

limitations que nous prévoyons d'étudier dans de futurs travaux, tels qu'une stratégie de repli lors de l'identification de concepts non pertinents.

En plus d'aider la recherche documentaire, notre approche pourrait être utilisée pour proposer des concepts intelligents et lisibles par un humain afin de l'aider durant sa recherche. Ceux-ci pourraient prendre la forme de nuages de mots ou d'entités (comme des pages Wikipédia par exemple). L'interaction d'un humain avec un système de recherche d'information pourrait ainsi évoluer de la simple reformulation de requête vers un affinage des concepts, ce qui permettrait de traiter directement le besoin d'information et non plus sa représentation exprimée par des mots-clés.

Remerciements

Ces recherches ont bénéficié du soutien financier de l'Agence Nationale de la Recherche (ANR 2010 CORD 001 02) en faveur du projet CAAS.

6. Bibliographie

AlSumait L., Barbará D., Domeniconi C., « On-line LDA : Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking », *Proceedings of ICDM*, 2008.

Andrzejewski D., Buttler D., « Latent topic feedback for information retrieval », *Proceedings of KDD*, 2011.

Arun R., Suresh V., Veni Madhavan C., Narasimha Murthy M., « On Finding the Natural Number of Topics with Latent Dirichlet Allocation : Some Observations », *Advances in Knowledge Discovery and Data Mining*, vol. 6118 of *Lecture Notes in Computer Science*, 2010.

Bai J., Nie J.-Y., Cao G., Bouchard H., « Using query contexts in information retrieval », *Proceedings of SIGIR*, 2007.

Bendersky M., Metzler D., Croft W. B., « Parameterized concept weighting in verbose queries », *Proceedings of SIGIR*, 2011.

Blei D. M., Ng A. Y., Jordan M. I., « Latent Dirichlet Allocation », *Journal of Machine Learning Research*, 2003.

Cao J., Xia T., Li J., Zhang Y., Tang S., « A density-based method for adaptive LDA model selection », *Neurocomputing*, 2009.

Chang Y., Ounis I., Kim M., « Query reformulation using automatically generated query concepts from a document space », *Information Processing & Management*, 2006.

Clarke C. L. A., Craswell N., Soboroff I., Cormack G. V., « Overview of the TREC 2010 Web Track », *Proceedings of the Nineteenth Text REtrieval Conference (TREC)*, 2010.

Clinchant S., Gaussier E., « Information-based models for ad hoc IR », *Proceedings of SIGIR*, 2010.

Cormack G., Smucker M., Clarke C., « Efficient and effective spam filtering and re-ranking for large web datasets », *Information Retrieval*, 2011.

Deveaud R., Bellot P., « Combinaison de ressources générales pour une contextualisation implicite de requêtes », *Actes de TALN*, 2012.

Egozi O., Markovitch S., Gabrilovich E., « Concept-Based Information Retrieval Using Explicit Semantic Analysis », *ACM Transactions on Information Systems*, 2011.

Finkelstein L., Gabrilovich E., Matias Y., Rivlin E., Solan Z., Wolfman G., Ruppin E., « Placing search in context : the concept revisited », *ACM Transactions on Information Systems*, 2002.

Graff D., Cieri C., « English Gigaword », *Philadelphia : Linguistic Data Consortium*, 2003.

Griffiths T. L., Steyvers M., « Finding scientific topics », *Proceedings of the National Academy of Sciences of the United States of America*, 2004.

Ingwersen P., « Polyrepresentation of information needs and semantic entities : elements of a cognitive theory for information retrieval interaction », *Proceedings of SIGIR*, 1994.

Jones K., *Retrieving Information Or Answering Questions ?*, British Library annual research lecture, British Library Research and Development Department, 1990.

Lavrenko V., Croft W. B., « Relevance based language models », *Proceedings of SIGIR*, SIGIR '01, 2001.

Lu Y., Mei Q., Zhai C., « Investigating task performance of probabilistic topic models : an empirical study of PLSA and LDA », *Information Retrieval*, 2011.

Metzler D., Bernstein Y., Croft W. B., Moffat A., Zobel J., « Similarity measures for tracking information flow », *Proceedings of CIKM*, 2005a.

Metzler D., Croft W. B., « A Markov random field model for term dependencies », *Proceedings of SIGIR*, 2005b.

Metzler D., Croft W. B., « Latent concept expansion using markov random fields », *Proceedings of SIGIR*, 2007.

Park L. A., Ramamohanarao K., « The Sensitivity of Latent Dirichlet Allocation for Information Retrieval », *Proceedings of ECML PKDD*, 2009.

Robertson S. E., Walker S., « Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval », *Proceedings of SIGIR*, 1994.

Sandhaus E., « The New York Times Annotated Corpus », *Philadelphia : Linguistic Data Consortium*, 2008.

Stock W. G., « Concepts and semantic relations in information science », *Journal of the American Society for Information Science and Technology*, 2010.

Wei X., Croft W. B., « LDA-based document models for ad-hoc retrieval », *Proceedings of SIGIR*, 2006.

White R. W., Bailey P., Chen L., « Predicting user interests from contextual information », *Proceedings of SIGIR*, 2009.

Yi X., Allan J., « A Comparative Study of Utilizing Topic Models for Information Retrieval », *Advances in Information Retrieval*, vol. 5478 of *Lecture Notes in Computer Science*, 2009.

Zhai C., Lafferty J., « A study of smoothing methods for language models applied to information retrieval », *ACM Transactions on Information Systems*, 2004.